A MEDICAL IMAGE PROCESSING AND ANALYSIS FRAMEWORK

A THESIS SUBMITTED TO
THE GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES
OF
MIDDLE EAST TECHNICAL UNIVERSITY

BY

ALPER ÇEVİK

IN PARTIAL FULFILLMENT FOR THE REQUIREMENTS
FOR
THE DEGREE OF MASTER OF SCIENCE
IN
BIOMEDICAL ENGINEERING

JANUARY 2011

A MEDICAL IMAGE PROCESSING AND ANALYSIS FRAMEWORK

submitted by **ALPER ÇEVİK** in partial fulfillment of the requirements for the degree of **Master of Science in Department of Biomedical Engineering, Middle East Technical University** by,

Prof. Dr. Canan Özgen
Dean, Graduate School of **Natural and Applied Sciences** _____

Prof. Dr. Mehmet Zülfü Aşık
Head of Department, **Dept. of Biomedical Engineering** _____

Prof. Dr. B. Murat Eyüboğlu
Supervisor, **Electrical and Electronics Engineering Dept., METU** _____

Prof. Dr. Kader Karlı Oğuz
Co-supervisor, **Dept. of Radiology, Hacettepe University** _____

Examining Committee Members:

Prof. Dr. Nevzat G. Gençer
Electrical and Electronics Engineering Dept., METU _____

Prof. Dr. B. Murat Eyüboğlu
Electrical and Electronics Engineering Dept., METU _____

Prof. Dr. Ayşenur Cila
Dept. of Radiology, Hacettepe University _____

Prof. Dr. Gerhard Wilhelm Weber
Institute of Applied Mathematics, METU _____

Assist. Prof. Yeşim Serinağaoğlu Doğrusöz
Electrical and Electronics Engineering Dept., METU _____

Date: 27.01.2011

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Name, Last Name: Alper, Çevik

Signature:

ABSTRACT

A MEDICAL IMAGE PROCESSING AND ANALYSIS FRAMEWORK

Çevik, Alper

M.Sc., Department of Biomedical Engineering
Supervisor : Prof. Dr. B. Murat Eyüboğlu
Co-supervisor: Prof. Dr. Kader Karlı Oğuz

January 2011, 111 Pages

Medical image analysis is one of the most critical studies in field of medicine, since results gained by the analysis guide radiologists for diagnosis, treatment planning, and verification of administered treatment. Therefore, accuracy in analysis of medical images is at least as important as accuracy in data acquisition processes.

Medical images require sequential application of several image post-processing techniques in order to be used for quantification and analysis of intended features. Main objective of this thesis study is to build up an application framework, which enables analysis and quantification of several features in medical images with minimized input-dependency over results. Intended application targets to present a software environment, which enables sequential application of medical image processing routines and provides support for radiologists in diagnosis, treatment planning and treatment verification phases of neurodegenerative diseases and brain tumors; thus, reducing the divergence in results of operations applied on medical images.

In scope of this thesis study, a comprehensive literature review is performed, and a new medical image processing and analysis framework - including modules responsible for automation of separate processes and for several types of measurements such as real tumor volume and real lesion area - is implemented. Performance of the fully-automated segmentation module is evaluated with standards introduced by Neuro Imaging Laboratory, UCLA; and the fully-automated registration module with Normalized Cross-Correlation metric. Results have shown a success rate above 90 percent for both of the modules. Additionally, a number of experiments have been designed and performed using the implemented application.

It is expected for an accurate, flexible, and robust software application to be accomplished on the basis of this thesis study, and to be used in field of medicine as a contributor by even non-engineer professionals.

Keywords: Medical image processing, image segmentation, image registration.

ÖZ

TIBBİ GÖRÜNTÜ İŞLEME VE ANALİZ UYGULAMA ÇATISI

Çevik, Alper

Yüksek Lisans, Biyomedikal Mühendisliği
Tez Yöneticisi : Prof. Dr. B. Murat Eyüboğlu
Ortak Tez Yöneticisi: Prof. Dr. Kader Karlı Oğuz

Ocak 2011, 111 Sayfa

Tıbbi görüntü analizi, verdiği sonuçlar doğrultusunda radyoloji uzmanlarına tanı, tedavi planı ve uygulanan tedavinin doğrulanması aşamalarında yol gösterici olduğundan, tıp biliminin en önemli çalışma alanlarından birisidir. Bu nedenle, tıbbi görüntülerin doğru analiz edilmesi, en az, veri elde etme sürecindeki doğruluk kadar önemlidir.

Tıbbi görüntülerin analiz edilebilmesi ve hedef özniteliklere ait nicel ölçüm bilgilerinin elde edilebilmesi için, görüntülerin bir dizi görüntü işleme tekniği uygulamasına tabi tutulması gerekmektedir. Bu tez çalışmasının ana amacı, tıbbi görüntüler üzerindeki birçok özniteliğin, sonuçlar üzerindeki kullanıcı bağımlılığı etkisinin en aza indirilmesiyle analiz edilmesi ve ölçümlenmesini mümkün kılacak bir uygulama çatısı meydana getirmektir. Tasarlanan uygulama, tıbbi görüntü işleme rutinlerini sıraya koyarak uygulamaya imkan vermeyi; radyoloji uzmanlarına, nörolojik dejeneratif hastalıklar ve beyin tümörlerinin tanı, tedavi planı ve tedavi doğrulama süreçlerinde destek olacak bir yazılım ortamı sunmayı; böylece, elde edilen sonuçlar üzerindeki varyasyonu düşürmeyi hedeflemektedir.

Bu tez çalışması kapsamında, kapsamlı bir literatür taraması gerçekleştirilmiş, ve işlemlerin otomatikleştirilmesinden ve gerçek tümör hacmi ve lezyon alanı gibi ölçümlerin yapılmasından sorumlu ayrık modüllere sahip, yeni bir tıbbi görüntü işleme ve analiz uygulama çatısı gerçekleştirilmiştir. Tamamen otomatikleştirilmiş bölütleme modülünün performansı, UCLA'daki Nörolojik Görüntüleme Laboratuarı'nın standartları doğrultusunda; tamamen otomatikleştirilmiş hizalama modülünün performansı ise, Normalize Edilmiş Çapraz Korelasyon ölçütü esas alınarak değerlendirilmiştir. Sonuçlar her iki işlem için de yüzde 90'ın üzerinde başarı oranı göstermiştir.

Bu tez çalışmasını temel alarak, doğruluk oranı yüksek, esnek ve sürdürülebilir, mühendis olmayan alan profesyonelleri tarafından da kullanılması mümkün bir yazılım ürününün gerçekleştirilmesi amaçlanmaktadır.

Anahtar kelimeler: Tıbbi görüntü işleme, imge bölütleme, imge hizalama.

to my family and friends...

ACKNOWLEDGEMENTS

I am heartily thankful to my supervisor, Prof. Dr. B. Murat Eyüboğlu, whose encouragement and support from the initial to the final level enabled me to develop an understanding of the subject. I owe my deepest gratitudes to Prof. Dr. Ayşenur Cila and Prof. Dr. Kader Karlı Oğuz for making it possible for me to complete this thesis. I would also like to thank Prof. Dr. Nevzat G. Gençer, Prof. Dr. Gerhard Wilhelm Weber and Assist. Prof. Dr. Yeşim Serinağaoğlu for their continuous support during my study.

I would like to thank Ajdan Küçükçiftçi, who has made available her support in a number of ways throughout last ten years of my life. I would like to show my gratitude to Dr. Güçlü Ongun, for bringing his deep knowledge into use. I would also like to thank Balkar Erdoğan and Berna Akıncı for their companionship. I wish to thank my lab-mate Kerim Kara, together with whom I have learnt almost everything involved in my study. I owe my deepest gratitudes to Assist. Prof. Dr. Altan Koçyiğit and to Dr. Serkan Kaygın for sharing know-how and experiences.

I would also like to thank all of my employers and colleagues in Hemosoft IT & Training Services Inc.. I am indebted my colleagues Yusuf Sayıta and M. Özge Kaya for their motivational support. I offer my regards and blessings to Ceren, Ersin, Alireza, Soner and all other friends from the laboratory.

Finally, and most importantly, I would like to thank my parents and my sister, Ayşe Çevik, Talat Çevik, and Melis Çevik who have supported and loved me for a lifetime.

TABLE OF CONTENTS

ABSTRACT iv

ÖZ vi

ACKNOWLEDGEMENTS ix

TABLE OF CONTENTS x

LIST OF TABLES xiv

LIST OF FIGURES xv

CHAPTERS 1

1. INTRODUCTION 1

1.1. Motivation 1

1.2. Objective 2

1.3. Main Contributions 3

1.4. Thesis Outline 4

2. BACKGROUND 6

2.1. Medical Image Processing 6

2.2. Image Smoothing and Restoration 8

2.2.1. Diffusion Filters 9

2.2.1.1. Linear Isotropic Diffusion 10

2.2.1.2. Nonlinear Isotropic Diffusion 11

2.2.1.3. Nonlinear Anisotropic Diffusion 12

2.2.2. Shock Filters 13

2.2.3. Total Variational Denoising 13
2.3. Image Segmentation 16
2.3.1. Medical Applications of Image Segmentation 17
2.3.2. Methods of Medical Image Segmentation 18
2.3.2.1. Mumford-Shah Segmentation 19
2.4. Image Registration 22
2.4.1. Medical Applications of Image Registration 23
2.4.2. Mathematical Background of the Rigid-Body Transformation 24
2.4.2.1. Forward Problem 25
2.4.2.2. Inverse Problem 27
2.4.2.3. Addition of Scaling Parameters 29
3. IMPLEMENTATION 31
3.1. Filtering Module 32
3.1.1. Linear Diffusion Filter 32
3.1.2. Perona-Malik Filter 33
3.1.3. Shock Filter 35
3.2. Image Segmentation Module 38
3.2.1. 2D Image Segmentation 38
3.2.2. Feature Area Computation 40
3.3. Image Registration Module 41
3.3.1. Registration with Known Correspondence 42
3.3.2. Fully-Automated Global Image Registration with Rigid-Body Transformation 44
3.3.2.1. Interpolation 44
3.3.2.2. Similarity Metric 45
3.3.2.3. Optimization 45

3.3.2.4. Transformation 46
3.4. 3D Medical Image Viewer and 3D Image Segmentation 48
3.4.1. 3D Medical Image Viewer 48
3.4.2. 3D Segmentation 50
3.4.3. Feature Volume Computation 50
4. PERFORMANCE EVALUATION 52
4.1. Validation of Image Segmentation Module 52
4.1.1. Metrics 52
4.1.1.1. Mean Preservation 52
4.1.1.2. Performance Measures 53
4.1.2. Results 56
4.2. Validation of Image Registration Module 59
4.2.1. Normalized Cross-Correlation Metric 59
4.2.2. Results 60
5. EXPERIMENTAL RESULTS 62
5.1. Image Smoothing and Restoration 62
5.1.1. Linear Diffusion Filter 63
5.1.2. Perona-Malik Filter 66
5.1.3. Shock Filter 69
5.2. 2D Image Segmentation and Area Computation 70
5.2.1. Analysis of Relation between Ambrosio-Tortorelli Energy and Input Parameters 79
5.3. 3D Image Segmentation and Volume Computation 81
5.3.1. Volumetric Segmentation 82
5.3.2. Volume Visualization 84
5.4. 2D Image Registration 86

5.4.1. Fully-Automated 2D Rigid Registration with Scale Parameters 86
5.5. 3D Image Registration 91
5.5.1. Fully-Automated 3D Rigid Registration with Scale Parameters 91
6. CONCLUSION 97
6.1. Conclusions 97
6.2. Future Work 99
REFERENCES 101
APPENDIX A. NUMERICAL SOLUTION OF AMBROSIO-TORTORELLI MINIMIZATION TO MUMFORD-SHAH ENERGY FUNCTIONAL..... 108

LIST OF TABLES

TABLES

Table 3.1 – Transformation types for cp2tform function [69] 43
Table 3.2 – Number of transformation parameters to be optimized 47
Table 4.1 – Performance evaluation results for simulated data 57
Table 4.2 – Performance evaluation results for real MRI data 59
Table 4.3 – Transformation parameters 61
Table 5.1 – Numerical information regarding segmentation of image with tumor 74
Table 5.2 – Numerical information regarding distance and area measurement 75
Table 5.3 – Numerical information regarding segmentation of image with MS lesions 78
Table 5.4 – 5 minimum points of SSD plot 80
Table 5.5 – Data regarding volume computation 84
Table 5.6 – Initial and final values for input parameters 87
Table 5.7 – Initial and final values for input parameters 96

LIST OF FIGURES

FIGURES

Figure 3.1 – General block diagram of the application 31
Figure 3.2 – Basic flowchart for the implementation of Ambrosio-Tortorelli segmentation 39
Figure 3.3 – Flowchart for the implementation of registration with known correspondence 42
Figure 3.4 – Fully-automated registration flowchart 44
Figure 3.5 – Visualization of transformation types, taken from reference [61] 47
Figure 3.6 – A screenshot of the 3D medical image viewer module in thresholding mode. 49
Figure 4.1 – Domain of metrics used for validation of segmentation, taken from reference [74]..... 54
Figure 4.2 – Original simulated image, taken from reference [74]..... 56
Figure 4.3 – Deformed simulated image (lower SNR), taken from reference [74]..... 56
Figure 4.4 – Segmented image..... 56
Figure 4.5 – Selected region 56
Figure 4.6 – User interface for area representation and performance evaluation 57
Figure 4.7 – Original MRI image, taken from reference [74] 58
Figure 4.8 – Segmented MRI image 58
Figure 4.9 – Ground truth segmentation, taken from reference [74] 58
Figure 4.10 – Selected Region..... 58

Figure 4.11 – Fixed image 60
Figure 4.12 – Moving image..... 60
Figure 4.13 – Aligned image 61
Figure 4.14 – Absolute difference of aligned and fixed images..... 61
Figure 5.1 – Original image, taken from reference [75]..... 63
Figure 5.2 – Image with salt & pepper noise..... 63
Figure 5.3 – Image with Gaussian white noise..... 63
Figure 5.4 – Image with both salt & pepper and Gaussian white noises..... 63
Figure 5.5 – Input image with salt & pepper noise 64
Figure 5.6 – Linear diffusion output (100 iterations)..... 64
Figure 5.7 – Absolute difference between input and output 64
Figure 5.8 – Linear diffusion output (10000 iterations)..... 64
Figure 5.9 – Mean value vs. iterations (LDF) 64
Figure 5.10 – Standard deviation vs. iterations (LDF)..... 64
Figure 5.11 – Entropy vs. iterations (LDF) 64
Figure 5.12 – Point spread function (PSF) for LDF 65
Figure 5.13 – Input image deformed by both Gaussian white noise and salt & pepper noise 67
Figure 5.14 – Non-linear isotropic filter output with contrast threshold = 0.005 67
Figure 5.15 – Non-linear isotropic filter output with contrast threshold = 0.001 68
Figure 5.16 – Output with hybrid application of LDF and Perona-Malik 68
Figure 5.17 – Blurred input for shock filter 69
Figure 5.18 – Successful output of shock filter 69
Figure 5.19 – Ringing effect caused by high weighting factor..... 69
Figure 5.20 – Unsuccessful output of shock filter - noisy image case..... 70
Figure 5.21 –Original brain MRI image with MS (multiple sclerosis)..... 71
Figure 5.22 – Original image with regions due to MS emphasized 71
Figure 5.23 – Original brain MRI image with edema [75]..... 72
Figure 5.24 – Selected ROI over image domain 72
Figure 5.25 – Segmented brain MRI image with edema 72

Figure 5.26 – Edge map of the segmentation 72
Figure 5.27 – ROI and boundaries 72
Figure 5.28 – Binary representation of selected area 72
Figure 5.29 – Statistical evolution regarding segmentation of image with tumor 74
Figure 5.30 – ROI and boundaries for edema region 75
Figure 5.31 – Binary representation of edema region 75
Figure 5.32 – Original brain MRI image with multiple sclerosis (MS) 77
Figure 5.33 – Selected ROI over image domain 77
Figure 5.34 – Segmented brain MRI image with MS 77
Figure 5.35 – Edge map of the segmentation 77
Figure 5.36 – ROI and boundaries 77
Figure 5.37 – Binary representation of selected area 77
Figure 5.38 – Data Fidelity Metric vs. Regularization & Edge Complexity (beta=1) 79
Figure 5.39 – Input image 80
Figure 5.40 – Reference segmentation 80
Figure 5.41 – Output corresponding to the 1st minimum 80
Figure 5.42 – Output corresponding to the 5th minimum 80
Figure 5.43 – A screenshot image of 3D Medical Image Viewer module 81
Figure 5.44 – 3D view of the segmented volumetric image 82
Figure 5.45 – Graphical representation of energy vs. iterations 83
Figure 5.46 – Volumetric representation of the selected object (tumor). 84
Figure 5.47 – Volume visualization example-1 85
Figure 5.48 – Volume visualization example-2 86
Figure 5.49 – Segmented brain MRI image with edema 87
Figure 5.50 – Segmented brain MRI without edema 87
Figure 5.51 – Trajectory of rotation angle 88
Figure 5.52 – Trajectory of translation parameters 89
Figure 5.53 – Trajectory of scaling parameters 89
Figure 5.54 – Evolution of minus normalized cross correlation 90

Figure 5.55 – Absolute difference between reference image and the initial version of moving image (input image) 91
Figure 5.56 – Absolute difference between reference image and the final version of moving image (corrected image) 91
Figure 5.57 – Trajectory of rotation parameters 92
Figure 5.58 – Trajectory of translation parameters 93
Figure 5.59 – Trajectory of scaling parameters 93
Figure 5.60 – Evolution of 3D minus normalized cross correlation 94
Figure 5.61 – Reference 3D image 94
Figure 5.62 – Initial 3D input image 94
Figure 5.63 – Corrected (registered) version of the moving image 95
Figure 5.64 – Visualization of absolute difference between reference and input images 95
Figure 5.65 – Visualization of absolute difference between reference and corrected images 95

CHAPTER 1

INTRODUCTION

1.1. Motivation

Medical imaging is the general name given to the group of techniques and processes developed for creating anatomical or functional images of human body (partially or as a whole), which are used for both clinical and scientific purposes. Medical image analysis is one of the most critical studies in field of medicine, since results gained by the analysis lead field professionals for diagnosis, treatment planning, and verification of administered treatment. Moreover, recent developments in medical imaging and medical image processing provided a significant reduction in the requirement for invasive intervention in treatment of various diseases or abnormalities.

Medical images require sequential application of several image post-processing techniques - such as restoration, regularization, segmentation and registration - in order to be used for quantification and analysis of intended features. These features may be specific parts of the image - like specified tissues, tumors, or lesions - as well as they may be any statistical property over the entire image domain or over parts of it.

Many application frameworks are developed in order to enable medical image data to be processed manually, semi-automated, or fully-automated by non-engineer field experts. However, effective use of many of these application domains requires remarkable amount of manual interaction. This situation creates several negations such as difficulty in use and diversity on acquired results.

In scope of this thesis work, a new medical image processing and statistical analysis framework has been developed based on a comprehensive review of modern image processing literature. Main motivation behind this thesis study was performing the analysis, design, implementation, and validation of a fast and robust system, which enables application of several medical image processing routines necessary for quantification and analysis of various image features in an effective way.

1.2. Objective

Main objective of this thesis study is to build up an application framework and a prototype application, which enables analysis and quantification of several features in medical images with minimized input-dependency over results. Intended application targets providing support for radiologists in diagnosis, treatment planning and treatment verification phases of neurodegenerative diseases and brain tumors; thus, reducing the divergence in results of operations applied on medical images.

Image filtering, image segmentation, and image registration operations are aimed to be handled in scope of the thesis. Also, development of a 3-dimensional (3D) volumetric medical image viewer is decided to be necessary in order to enable selection of "*region of interest*" (ROI) for volumes and regions to be quantified, and visualization of processed volume images.

Construction of mentioned application framework undoubtedly requires a wide exploration of the image processing literature. Literature review presented in CHAPTER 2 (*Background*) is determined to involve modern approaches, as well as the traditional methods which are already in practical use. Mathematical background is aimed to be used in an optimization perspective in automation of purposed operational modules.

1.3. Main Contributions

A comprehensive review of image processing literature is made, and image processing routines and mathematical relations among them are deeply investigated. Our review directs reader to related previous work on each topic, and presents detailed mathematical background behind numerical solutions to problems and implementation of chosen methods.

A prototype tool is constructed using the data structures developed in scope of the application framework. This tool is composed of 4 modules, each one of which has several functions and capabilities. These modules are:

- Image Filtering Module,
- Image Segmentation Module,
- Image Registration Module, and
- 3D Image Viewer.

Detailed information on implementation, validation, and experiments done using these modules is introduced in related chapters.

Image filtering module enables application of some techniques for restoration and regularization on the images. “*Linear Diffusion Filter*”, “*Perona-Malik*

Filter", and "*Shock Filter*" are programmatically implemented and plugged into the module in this extent.

Image segmentation module involves Mumford-Shah based automated segmentation of images, and object selection with automated region growing. Segmentation operation is followed with 2D cross-sectional area and 3D volume computations for selected objects.

Image registration with known correspondence is provided with several transformation models. Also, a top-down fully-automated procedure is proposed and performed for global registration with rigid-body transformation of 2D and 3D images.

A 3D image viewer, which shows *axial*, *sagittal*, and *coronal* projections of volumetric medical images, is built as a helper for 3D segmentation and feature quantification. A volumetric rectangular ROI can be selected and operated via using this module.

Although experimental work is mainly centered upon 2D and 3D (volumetric) brain MRI images, it is not obligatory for application to be operated with mentioned modality. Any "DICOM" images and MATLAB® data files with ".mat" extension can be opened and processed within the framework.

1.4. Thesis Outline

First chapter (*Introduction*) gives a brief explanation of the motivation behind this study and the main objectives inside the scope of this thesis. A summary of performed work and resulting contributions is presented in Section 1.3, and Section 1.4 gives the outline of the thesis.

Following chapter - Chapter 2 - is entitled as "*Background*", and written in order to give the mathematical background behind this thesis. Previous work related to the topics of our study is also introduced in this chapter. First

section (Section 2.1) gives an introduction to the requirement for use of image processing in field of medicine. Following sections of the chapter are organized in an aspect of classification of main image processing procedures, and the subsections give medical applications and mathematical background regarding each topic.

Chapter 3 (*Implementation*) mainly deals with the problems and solutions in construction phase of the application. Sections are categorized with respect to the above named (Section 1.3) modules of the applications. For each one of the modules, numerical solutions of selected models are presented, and programmatic perspectives are tried to be put across.

Validation methods used for the performance evaluation of parts of the application are explained in Chapter 4. Metrics used for evaluation of success in image registration and image segmentation operations are defined, and validation results are presented.

Following chapter - Chapter 5 - presents the experiments done on sample images ad image sets, their results and interpretations on these results. Again, division of sections is based-on operational modules of the application, and subsections are categorized in accordance with functions of these modules.

Chapter 6 is the final chapter of, and a conclusion regarding this thesis study is introduced here. Conclusion part tries to give answers to some questions, such as:

- “What is proposed?”,
- “What is accomplished?”, and,
- “What can be done in the future?”.

CHAPTER 2

BACKGROUND

Medical imaging is the general name for the widely-used techniques developed in order to create images of human body for medical purposes. As acquired images could involve complete human body, they can span it partially. Medical imaging data is used for revealing normal or abnormal physiological and anatomical structures. Medical imaging techniques are also employed in diagnosis and treatment planning processes of patients suffering from many health problems. Professionals from field of medicine make use of medical imaging data in order to guide or avoid medical intervention.

2.1. Medical Image Processing

Image processing is a subfield of signal processing, for which the input signal is an image and the outputs are again an image and/or various parameters defining the characteristics of the image and applied operations. Medical image processing is applied on the images acquired by medical imaging techniques, such as CT, Ultrasonography, PET, SPECT, MRI, fMRI and

NMR spectroscopy. Medical images are post-processed for many purposes, such as denoising, restoration, segmentation, registration, and 2D/3D visualization [**1**,**2**].

Quantitative analysis of medical images is crucial for diagnosis and prognosis stages of many diseases and abnormalities. Quantification of radiographic information includes various features such as linear measurements, estimation of cross section and surface areas, volume quantization, estimation tissue density, monitoring tumor growth, verification of treatment, and comparison of patient's data with anatomical atlases [**3**,**4**].

Medical image data is exposed to degradations and/or deformations during data acquisition processes. For instance, MRI intensity inhomogeneities occur subject to RF coil imperfections or problems associated with acquisition sequences [**5**]. Hence, quantification of medical image information for analysis requires sequential application of several image processing operations. These operations can be classified in three main groups which are smoothing and restoration of images, segmentation of images, and registration of images. Each step of this workflow necessitates user interaction at varying levels. Manually operating, semi-automated and even automated systems receive numerical, vectorial or optional inputs from the user, defining the instantaneous or future behavior of running operations. Diversity in the number of user interaction points may cause complete or partial failure of the operations, or irregularities or instabilities in the acquired results. Even if the processes are completed and reasonable results are obtained, sets of these results depending on different users and/or different time instants may have high standard deviation values.

As briefly stated in the introduction part, main objective of this thesis is to build an image processing framework which enables application of addressed sequential medical image processing operations with minimum diversity depending on user interaction in the results. Following sections of this chapter introduce previously indicated classes of operations used in

medical image processing in the aspects of mathematical methodology and applications in the literature.

2.2. Image Smoothing and Restoration

Filtering plays a significant role in image enhancement. Diffusion filters are well understood and used as powerful tools in image analysis for several decades. Diffusion filters can be grouped as linear and nonlinear, or isotropic and anisotropic diffusion filters.

Formulation of diffusion filtering processes with partial differential equations (PDEs) created a solid backbone for a common framework - a top-down methodology - for scale-space analysis. References [**6**] and [**7**] give interesting relations between biological mechanisms behind vision and scale-space analysis. Morel and Solimini [**8**] introduced a deep mathematical and physical understanding of this methodology for multiscale image smoothing and restoration. Mentioned methodology involves four main steps, which are:

- Mathematical modeling of the energy functional to be optimized (minimized or maximized according to the nature of the model) ,
- Derivation of the time and space dependent PDEs (Euler-Lagrange equations) using “Calculus of Variations”,
- Intrinsic/extrinsic solution of PDEs in time and space (or frequency) domain, and
- Implementation of numerical solution on the input image (or image set).

In this aspect, images are treated as continuous signals, carrying the information of gray levels depending on space variables (pixel or voxel positions in discrete domain) and time variable. Time variable is used analogous to evolution of image under physical diffusion concept.

This section is presented for a review of diffusion filters and other image enhancement methods developed in scope of variational approach.

2.2.1. Diffusion Filters

Fick's 2nd law of diffusion is known as:

$$\partial u/\partial t = \nabla \cdot (C \nabla u). \qquad \textbf{(2.1)}$$

This equation is called "diffusion equation" and is valid for explaining many physical transportation models. Since it is appropriate for heat transfer, it is also called "heat equation". In image processing context, u is the image signal, and C is the diffusion coefficient. For the time being, it is convenient to assume C as a scalar quantity. $C\nabla u$ term creates a flux, and combined with the continuity equation, divergence of this flux gives $\frac{\partial u}{\partial t}$ with preservation of the mass.

Diffusion filtering guarantees that the operation does not change mean value of the pixel (or voxel) intensities. In other words, average grey value remains constant independent from the total number of iterations. This has been ensured by employing Neumann (or second-type) boundary condition [**9**] on the edges of the image, which creates a reflection of pixels on boundaries

beyond the edges. Consequently, neither sources nor sinks can appear on the image domain.

2.2.1.1. Linear Isotropic Diffusion

Discrete solution of Equation **(2.1)** is given in Equation **(2.2)**:

$$u_{i,j}^{k+1} = (C\Delta t/h^2)\left(u_{i+1,j}^{k} + u_{i-1,j}^{k} + u_{i,j+1}^{k} + u_{i,j-1}^{k}\right) + (1 - 4C\Delta t/h^2)u_{i,j}^{k} \quad . \tag{2.2}$$

Here, $u_{i,j}^{k}$ denotes the intensity of pixel in i^{th} row and j^{th} column of the image matrix u at k^{th} iteration. Δt is the temporal and h is the spatial step size. Inequality **(2.3)** shows the stability constraint over Equation **(2.2)**:

$$(1 - 4C\Delta t/h^2) \geq 0. \tag{2.3}$$

Therefore, diffusion weighting factor $C\Delta t/h^2$ should be smaller than or equal to 0.25, and constant over the image domain over time.

Linear diffusion filter non-directionally blurs the original image and operates as a low-pass filter. As time (number of iterations) goes to infinity, image evolves into a uniform gray image with the mean value of pixel intensities of original image signal at each one of the pixels.

There exists an analogy between convolution of the image with a Gaussian bell-shaped curve, and linear diffusion operation. Mathematical relations between Gaussian smoothing parameters and linear diffusion parameters are reviewed in detail at reference [**10**].

With the intention of understanding linear diffusion operation better, the process can be visualized as moving a circle with constant radius, which spreads the information on the central point towards the edge points. Rate of diffusion ($C\Delta t/h^2$ term in Equation **(2.2)**) is proportional with the radius (distance from the central point); therefore, it is constant for each one of the image pixels on which the circle is located consequently. As long as shape of the virtual moving object remains circular, diffusion process is independent from direction; therefore, it can be said to be isotropic.

2.2.1.2. Nonlinear Isotropic Diffusion

Perona and Malik [**11**] proposed the first non-linear diffusion filter model, with modifying linear isotropic diffusion expression given in Equation **(2.1)**. In their model, diffusion constant C is replaced with a scalar function $g(|\nabla u|^2)$, where:

$$g(|\nabla u|^2) = 1/(1 + |\nabla u|^2/\lambda^2) \qquad (\lambda > 0). \qquad \textbf{(2.4)}$$

According to the Perona-Malik model given by Equation **(2.4)**, magnitude of the diffusion coefficient decreases with increasing gradient on the image. Variable λ is called contrast threshold, and controls the speed of change in diffusion coefficient. Diffusion weight is maximized at zero-gradient points,

and at approaches to zero as gradient approaches infinite. This provides preservation of edges on the image while blurring other parts of it.

After replacing C with $g(|\nabla u|^2)$, Inequality **(2.3)** is valid as the stability condition over Perona-Malik model. As time goes to infinity, original image evolves into a uniform gray image just like it does in linear diffusion case.

If the progress is compared with the previously mentioned moving circle example, the only difference from linear isotropic diffusion case is the radius of the circle being varying with position of center, due to gradient at that point in space. Perona and Malik gave the name “Anisotropic Diffusion” to their diffusion model when they came up with it. Unfortunately, it is not anisotropic since the rate of diffusion does not change with the diffusion direction. Although Perona and Malik have corrected later on, this common denomination is still present in many scientific resources.

2.2.1.3. Nonlinear Anisotropic Diffusion

Going ahead with the moving circle annotation, anisotropic diffusion can be explained by the shape properties of the circle being modified. Purpose of the filtering process is not only decreasing the diffusion rate on the edges, and also enhancing them with increasing the diffusion rate along the edges. Therefore, a moving circle cannot handle the operation. What is necessary is an elliptic object with varying eccentricity.

Equation **(2.1)** should be modified so that scalar valued C is turned into a positive definite symmetric matrix. This diffusion tensor can be designed according to the objective. Edge enhancing and coherence enhancing models and numerical solution methods are introduced in detail on references [**10**,**12**,**13**].

2.2.2. Shock Filters

Shock filters are developed with the idea of assuming the initial image as the output of a diffusion operation. Therefore, modeling of a shock filter involves solution of a differential equation satisfying a maximum principle, reversing the time variable t of Equation **(2.1)**. Upwind scheme is applied in discretization of the differential equation.

Shock filters are generally used in image restoration purposes. Shock filtering process gives better results with blurred images. However, it should not be preferred applying shock filter on images having "*salt & pepper noise*" since this class of filters have high sensitivity to gradient over signals.

Osher and Rudin [**14**] introduced a total-variation preserving shock filter. Alvarez and Mazolla [**15**] defined a class of filters combining shock filters and anisotropic diffusion in order to eliminate the noise-amplification effect of classical shock filters. More recent approaches employing complex diffusion, hybrid procedures, and clustering methods are presented in references [**16**,**17**,**18**,**19**].

2.2.3. Total Variational Denoising

"Total variation" (TV) on a subset of a function is the largest sum of "variations" for any subdivision of that subset. The total variation of a continuously differentiable function between points a and b can be given in one dimension as:

$$V_b^a(u(x)) = \int_a^b |u'(x)| dx. \qquad \textbf{(2.5)}$$

Equation **(2.5)** can be rewritten as given below for our case (2D images):

$$V(u(x,y),R) = \int_R |\nabla u(x,y)|. \qquad \textbf{(2.6)}$$

In Equation **(2.6)**, R denotes the image domain, x and y are the spatial coordinates of the points on the image domain.

Image restoration is defined as the process to compensate for or to undo the defects which degrade the image. This degradation may come in forms of blurring and noise.

Image function to be restored and regularized is generally defined as given in **(2.7)**:

$$z = u + \varepsilon. \qquad \textbf{(2.7)}$$

In Equation **(2.7)**, z, u, and ε are all 2D image signals and they are the expressions for observed image (with noise), true (desired) image, and error distribution, respectively. There are several methods [**20**,**21**] developed in order to be able to reconstruct u from z, including total variation (TV) minimization, which is the subject of this section. It is observed that TV minimization methods are effective for recovery of blocky (discontinuous) image data [**22**].

In [**23**], Rudin, Osher, and Fatemi considered the constrained minimization problem:

$$\text{minimize}_u \int_R |\nabla u| dx, \quad \textbf{(2.8)}$$

subject to Equation **(2.9)**

$$\|u - z\|^2 = \sigma^2. \quad \textbf{(2.9)}$$

In Equation **(2.9)**, σ gives a measure of distortion on the image.

In [**22**], Vogel and Oman consider the problem of minimization of TV-penalized least squares functional:

$$f(u) = \frac{1}{2}\|u - z\|^2 + \alpha \int_R \sqrt{|\nabla u|^2 + \beta^2}\, dx\,. \quad \textbf{(2.10)}$$

In Equation **(2.10)**, the first term is generally called "*data fidelity term*" and defines the measure of goodness to fit the data, and the second term depends on the total variation over the image signal and generally named as "*regularization term*". Positive parameter α determines the ratio between the efforts given to minimize these two terms. Here, β is a real number, which is plugged into the term in order to inhibit the regularization term from vanishing to zero.

Minimization of the functional given in Equation **(2.10)** provides good results in restoration of blocky images, because sharp gradients of these discontinuity regions help the process to be edge preserving. Various algorithms for numerical solution of TV minimization problem are introduced in references [**24**,**25**,**26**,**27**].

2.3. Image Segmentation

Recalling Equation **(2.7)**, the terms u, and ε can be redefined as follows:

$\boldsymbol{u}$: "*Cartoon*" image component composed of distinct smooth sub-regions which are separated from each other with respectively high gradient boundaries, and have ideally zero gradients inside the boundaries.

$\boldsymbol{\varepsilon}$: Summation of relatively small-amplitude high frequency signal component called "*texture*", and any other form of residual signal components, altogether called "*noise*".

Both from the perspectives of clustering and classification, image segmentation is an inverse problem solution process, of which the main goal is to "*cartoonize*" the image; in other words, to extract the cartoon component u from the original image z, which is composed of several - semantically meaningful - distinct partitions.

Above definition of image segmentation can easily be associated with the definitions of image denoising and restoration in terms of parameters and objectives. For the entire image processing methodology including those problems that are recently mentioned, additionally image registration and image inpainting subjects, this association is the key concept for understanding nature of image processing science. In other words, for the problem solution methods which do not employ prior information on image features like edges, shapes, or correspondence points, it may not be possible to disintegrate image processing procedures straightforwardly.

2.3.1. Medical Applications of Image Segmentation

Medical imaging techniques are used in order to gain information on specific organs or parts of human body, physiological abnormalities such as tumors and cysts, or any other structures like bone, cartilage, and vessels. In general, analysis of medical images requires segmentation of the images.

Reference [**28**] gives a comprehensive review of application areas and methods in medical image segmentation. Image segmentation is methods are in practical use in field of radiology in order to assist or automate multiple procedures. Significant roles of image segmentation applications in:

- Anatomical research on regular body structure,
- Quantification of tissues in several metrics like distance, cross-sectional and surface area, and volume,
- Classification of several special tissues like white matter and gray matter of brain,
- Diagnostic radiology,
- Localization of abnormalities, malfunctions, and pathologies,
- Prognosis and treatment planning,
- 2D and 3D registration of imaging data acquired at various times,
- Computer-aided surgery, and
- Volume correction in functional imaging data,

can be observed from references [**3**,**29**,**30**,**31**,**32**,**2**].

2.3.2. Methods of Medical Image Segmentation

Various traditional methods and modern approaches exist in field of medical image segmentation. Straightforward binary segmentation methods such as “thresholding” and “region growing” can be preferred on account of the fact that they are both easy implemented and fast operated procedures. However, these methods suffer from considerable user-dependency over the results of operations.

Thresholding method requires numerical threshold value (or multiple threshold values for multi-thresholding situation [**33**]) as input in order to create partitions belonging to different level of intensities. Similarly application of region growing method is in need of determination of a seed point and intensity similarity metric. Split and merge algorithms are mathematically related to region growing, however they do not require seed point specification.

An inclusive survey on classification methods such as use of Bayes (maximum likelihood) classifier, and clustering methods like fuzzy c-means algorithm, K-means (ISODATA) algorithm, and the expectation-minimization (EM) algorithm is presented in reference [**28**]. One disadvantage of employing classification methods for segmentation is that they require manual interaction for collection of the training data. Although clustering is an unsupervised (self-training) methodology, it requires initial parameters in case of segmentation applications.

Far-reaching information on several other medical image segmentation methods such as the watershed algorithm [**34**,**35**], artificial neural networks [**36**], Markov random field models [**37**], level set segmentation methods [**38**], deformable active contour models [**39**,**40**], model-fitting methods [**41**], atlas-based methods [**42**,**43**], and examples of hybrid implementation of mentioned methods [**44**,**45**,**46**,**47**] exist in the literature.

As indicated previously, one of the main objectives of this thesis work is to minimize the effects of manual interaction, and to narrow down the range of attained results with minimum inputs. Therefore, Mumford-Shah based segmentation methods are explored deeper in scope of this study. Justification of this approach will be clearer with mathematical explanation in the following subsection.

2.3.2.1. Mumford-Shah Segmentation

Mumford and Shah [**48**] defined the segmentation problem as minimization of the energy functional:

$$E = \beta \iint_R (u - z)^2 + \alpha \iint_{R-B} |\nabla u|^2 + \iint_R length(B). \qquad \textbf{(2.11)}$$

Equation **(2.11)** defines an energy functional which is formed by summation of three terms. Using the same convention with Equation **(2.7)**, z and u are the original image and the segmented image, respectively. R denotes the image domain. Hence, $\iint_R (u - z)^2$ term multiplied by a weighting factor of β (a positive real number), gives a measure of dissimilarity of the segmented image to the original data. Therefore, the first term on the right hand side of Equation **(2.11)** is usually called as "*data fidelity term*".

B defines the set of points which compose the boundaries of the segmented image. As a consequence, $R - B$ domain over which $|\nabla u|^2$ is integrated is the set of non-boundary points in the segmented image. $|\nabla u|^2$ is obviously an inverse measure for smoothness over the distinct partitions separated with image boundaries. Ideally (for a pure cartoon image), second term on the

right hand side of Equation **(2.11)** is equal to zero. Multiplied by a positive real weighting factor α, the term is ordinarily named as "*regularization term*".

Third term on the right side of Mumford-Shah energy functional gives the total length of the boundaries over the segmented image. Boundaries are the features which create high gradient over the segmented image domain. Observing from linear diffusion perspective, value of the third term approaches to zero with each iteration, and finally vanishes. This banishes the image from its original state. Therefore, minimization of the third term contradicts with minimization of the data fidelity term. Hence, minimization of the Mumford-Shah energy functional as a whole, transfers the image into an equilibrium state between restoration and regularization.

According to Mumford and Shah, solution of equation for minimization of the energy functional defined by Equation **(2.11)** gives the ideal segmented image (u) for the original image z. However, numerical solution of this problem is not a straightforward issue because of the existence of several local minimum values, non-regularity of the edge term, and discontinuity in the domain of the problem.

Several procedures are proposed for minimization of Mumford-Shah energy functional in references [**49**,**50**,**51**,**52**,**53**,**54**,**55**,**56**,**57**,**58**] and [**59**]. Given references include utilization of simulated annealing, graph cut algorithms, level set (spline) methods, convex relaxation approaches, and finite-difference discretization for segmentation based on minimization of Mumford-Shah functional. Each of these algorithms work well in practice although they have various drawbacks, such as converging to local minimums, not allowing open boundary formation, and excessiveness of the number of iterations to reach a convergence criterion.

In [**60**], Ambrosio and Tortorelli proposed an approximation for the Mumford-Shah energy functional (Equation **(2.11)**), which allows formation of open boundaries, in this sense, is more appropriate with the nature of the original energy, respectively. They proposed to replace the edge-set term by means

of defining a 2D function v and designed the phase field energy term given below:

$$L_{v,\rho} = \iint_R \{\rho|\nabla v|^2 + (1-v)^2/4\rho\} d\boldsymbol{x}. \quad \textbf{(2.12)}$$

In Equation **(2.12)**, ρ denotes a small positive real number and $\boldsymbol{x}$ is the coordinate vector over 2D image domain. Although function v has no explicit mathematical definition, it can be defined as given by Equation **(2.13)**:

$$\lim_{\rho \to 0} \frac{1}{2} \iint \{\rho|\nabla v|^2 + v^2/\rho\} = length(B). \quad \textbf{(2.13)}$$

If the edge term $\iint_R length(B)$ in original Mumford-Shah energy functional is replaced with the phase field energy term $L_{v,\rho}$ given in Equation **(2.12)**, and the resulting equation is reorganized such that all terms of integration fall onto the same domain, we end up with the Ambrosio-Tortorelli approximation of Mumford-Shah energy functional, E_{AT} given in Equation **(2.14)**:

$$E_{AT} = \iint [\beta(u-z)^2 + \alpha|\nabla u|^2(1-v)^2 + (1/2)\{\rho|\nabla v|^2 + (v^2/\rho)\}] d\boldsymbol{x}. \quad \textbf{(2.14)}$$

Equation **(2.14)** has discrete numerical solutions for u (segmented image) and v both of which can be factorized in PDE form using implicit scheme.

Within the scope of this thesis work, segmentation subject is dealt mainly around the concept of solution, implementation and observation of the behavior of the aforementioned method.

2.4. Image Registration

Image registration can be defined as the iterative process of searching for the best mathematical transformation model, which aligns a 2-dimensional (2-D) or 3-D image data to another 2-D or 3-D image data. Main objective of the process is to search for the transformation matrix, which gives the optimal value according to the decided similarity measure between two image data. Input image which is processed to be aligned to another image is called as "*moving image*". Reference image according to which the moving image is aligned is named as "*fixed image*". Mathematical similarity measure between moving and fixed images is called "*metric*".

Image registration applications can be classified according to several different aspects [**4**]. For example, we can categorize the methods as "*feature-based*" or "*area-based*" in terms of the features to be aligned, as "*full-automated*", "*semi-automated with known correspondence*", or "*manually operated*" in terms of degree of interactive guidance, and as "*rigid-body transformation*" or "*elastic transformation*" in terms of the determined constraints over the desired transformation model.

Following subsection gives a brief review of applications of image registration in field of medical image processing.

2.4.1. Medical Applications of Image Registration

Utilization of image registration techniques is crucial for the accuracy of analysis in field of medical imaging. Medical image registration is widely used for several purposes; such as, alignment of anatomical and functional images from different imaging modality classes, fusion of functional and anatomical images of a specific subject, analysis of time series of image data belonging to a particular subject, creation of ensemble maps for specific abnormality in specific body parts of multiple subjects, and refinement of volumetric data by degrading the negative effect of patient movement during acquisition. Additionally, treatment verification based on comparison of medical images acquired before and after the medical procedure always involves registration of medical images.

Two comprehensive reviews including detailed classification of methods and information on applications are presented in references [**61**,**62**].

Medical image registration is based upon finding correspondences between two images to be spatially aligned. Techniques of finding correspondences can be classified into two main groups which are feature-based and intensity-based methods [**62**]. "*Head-and-hat*" algorithm [**63**], "*Chamfer matching*" method [**64**], and "*iterative closest point*" (ICP) algorithm [**65**] may be given as the most popular examples of feature-based image registration techniques. Major drawback of feature-based image registration techniques is that the accuracy of registration is limited with the accuracy of feature extraction operations such as image segmentation. Intensity-based methods are based on utilization of intensity similarity measures. Some examples to these metrics can be stated as "*sum of squared differences*", "*cross-correlation*", and "*variance ratios*" of intensity values [**66**]. Additionally, there exist intensity-based methods based on information theory; such as "*joint histogram estimation*" [**67**] and "*maximization of mutual information (MI)*" [**68**]. Compared with the feature-based methods, intensity-based methods make

use of much more - generally all - of the available information on the image domain.

Brain is bounded by the skull. Also, long-term changes in the anatomical structure of brain are non-effective regarding the shortness of the time interval necessary to acquire medical images. Moreover, in prognosis of neurodegenerative diseases like multiple sclerosis (MS) and analysis of brain tumors, image registration aims to preserve changes in dimensions of lesions and tumors over time. In other words, the moving image should be registered to the fixed one globally. Regarding given reasons, global implementation of “rigid-body transformation model” on the moving image domain is appropriate for the registration of brain MRI images. Thus, following subsection deals with the mathematical principles behind the rigid-body transformation.

2.4.2. Mathematical Background of the Rigid-Body Transformation

Rigid transformation model consists of translation followed by a rotation operation. In case of mono-modal or multi-modal registration of brain MRI images, scale parameters are inserted to the model. Addition of the scaling operation turns the model actually into an “affine transformation” model, except from the capability to shear the image features. Although any transformation is required to keep the Euclidean distances between any two points in the image domain same in order to be respected as rigid, mentioned model (consisting of rotation, translation and scaling) is often referred as rigid transformation in the literature [**62**].

Mathematical expression for the transformation applied on an image is presented in Subsection 2.4.2.1 as the “Forward Problem”. Principles of reversing the transformation in order to align the moving image to the fixed image are introduced in the next Subsection 2.4.2.2, “Inverse Problem”.

Modifications implemented for addition of scaling parameters to the model are explained in Subsection 2.4.2.3, namely “Addition of Scaling Parameters”.

2.4.2.1. Forward Problem

Rigid transformation (without scaling) can be formulated as:

$$\mathbf{T}(\mathbf{x}) = \mathbf{R}\mathbf{x} + \mathbf{t}. \tag{2.15}$$

In Equation **(2.15)**, $\mathbf{T}$, $\mathbf{R}$, and $\mathbf{t}$ are all in matrix form, and denote rigid transformation matrix, rotation matrix (Equations **(2.16)** for 2D space and **(2.17)** for 3D space), and translation vector (Equations **(2.18)** for 2D space and **(2.19)** for 3D space), respectively. $\mathbf{x}$ is the coordinate point vector, which is also expressed in matrix form in Equations **(2.20)** for 2D space and **(2.21)** for 3D space:

$$\mathbf{R_{2D}} = \begin{bmatrix} \cos(\phi) & \sin(\phi) \\ -\sin(\phi) & \cos(\phi) \end{bmatrix} \tag{2.16}$$

$$\mathbf{R_{3D}} = \begin{bmatrix} \cos(\beta)\cos(\gamma) & \cos(\alpha)\sin(\gamma) + \sin(\alpha)\sin(\beta)\cos(\gamma) & \sin(\alpha)\sin(\gamma) - \cos(\alpha)\sin(\beta)\cos(\gamma) \\ -\cos(\beta)\sin(\gamma) & \cos(\alpha)\cos(\gamma) - \sin(\alpha)\sin(\beta)\sin(\gamma) & \sin(\alpha)\cos(\gamma) + \cos(\alpha)\sin(\beta)\sin(\gamma) \\ \sin(\beta) & -\sin(\alpha)\cos(\beta) & \cos(\alpha)\cos(\beta) \end{bmatrix} \tag{2.17}$$

In Equation **(2.16)** ϕ is the rotation angle in 2D space. Similarly, α, β, and γ parameters in Equation **(2.17)** denotes rotation angles around x, y, and z axes in 3D space. Centre of rotation is assumed as the central point of the images for both cases.

$$\mathbf{t_{2D}} = (t_x, t_y)^T \quad \textbf{(2.18)}$$

$$\mathbf{t_{3D}} = (t_x, t_y, t_z)^T \quad \textbf{(2.19)}$$

Translation can be defined as the linear movement of the image over the coordinate axes as a whole. In Equations **(2.18)** and **(2.19)**, mathematical expressions for 2D and 3D translation are given. t_x, t_y, and t_z give the projections of displacement vector over corresponding coordinate axis.

$$\mathbf{x_{2D}} = (x, y)^T \quad \textbf{(2.20)}$$

$$\mathbf{x_{3D}} = (x, y, z)^T \quad \textbf{(2.21)}$$

In Equations **(2.20)** and **(2.21)**, x, y, and z stand for Cartesian coordinates of the points in the moving image domain, for the 2D and 3D space cases, respectively.

2.4.2.2. Inverse Problem

Global image registration with rigid-body transformation is an optimization problem of searching the appropriate values for each of the transformation parameters, which optimize the similarity metric between the reference image and transformed version of the moving image. Thus, it requires solution of an inverse problem. In this scope, we assume moving image as an output of a transformation operation applied on the target image. Therefore, an inverse transformation model should be designed and appropriate values should be assigned for model parameters.

$$\mathbf{T_{rigid}} = \mathbf{T_{rotation}}\mathbf{T_{translation}} \quad \textbf{(2.22)}$$

Rigid transformation matrix can be expressed as multiplication of rotation and translation matrices, definitions of which are given by Equations **(2.23)** - **(2.25)** (rotation and translation matrices for 2D case) and **(2.24)** - **(2.26)** (rotation and translation matrices for 3D case):

$$\mathbf{T_{rotation2D}} = \begin{bmatrix} \cos(\emptyset) & \sin(\emptyset) & 0 \\ -\sin(\emptyset) & \cos(\emptyset) & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad \textbf{(2.23)}$$

$$\mathbf{T_{rotation3D}} = \begin{bmatrix} \cos(\beta)\cos(\gamma) & \cos(\alpha)\sin(\gamma) + \sin(\alpha)\sin(\beta)\cos(\gamma) & \sin(\alpha)\sin(\gamma) - \cos(\alpha)\sin(\beta)\cos(\gamma) & 0 \\ -\cos(\beta)\sin(\gamma) & \cos(\alpha)\cos(\gamma) - \sin(\alpha)\sin(\beta)\sin(\gamma) & \sin(\alpha)\cos(\gamma) + \cos(\alpha)\sin(\beta)\sin(\gamma) & 0 \\ \sin(\beta) & -\sin(\alpha)\cos(\beta) & \cos(\alpha)\cos(\beta) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

(2.24)

In Equation **(2.23)**, the variable $\emptyset$ stands for the rotation angle. α, β, and γ parameters in Equation **(2.24)** denotes rotation angles around x, y, and z axes in 3D space. Translation matrix is constructed placing the vectorial expressions in forward case as the first rows of last column of an identity matrix. Dimensions of translation matrices are 3x3 for 2D case, and 4x4 for 3D case.

$$\mathbf{T}_{\mathbf{translation2D}} = \begin{bmatrix} 1 & 0 & t_x \\ 0 & 1 & t_y \\ 0 & 0 & 1 \end{bmatrix} \quad \textbf{(2.25)}$$

$$\mathbf{T}_{\mathbf{translation3D}} = \begin{bmatrix} 1 & 0 & 0 & t_x \\ 0 & 1 & 0 & t_y \\ 0 & 0 & 1 & t_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad \textbf{(2.26)}$$

Equations **(2.27)** and **(2.28)** gives applied mathematical transformation model in opened form for 2D and 3D cases, respectively. For 2D case, there exist 2 parameters for translation and single parameter for rotation. 3D case has 6 degree of freedom, consisting of 3 translation and 3 rotation parameters. Detailed derivation and polar form of the model can be reviewed from reference [**2**].

$$\mathbf{T}_{\mathbf{rigid2D}}(\mathbf{x}) = \begin{bmatrix} \cos(\emptyset) & \sin(\emptyset) & 0 \\ -\sin(\emptyset) & \cos(\emptyset) & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & t_x \\ 0 & 1 & t_y \\ 0 & 0 & 1 \end{bmatrix} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix} \quad \textbf{(2.27)}$$

$$\mathbf{T_{rigid3D}}(\mathbf{x}) =$$

$$\begin{bmatrix} \cos(\beta)\cos(\gamma) & \cos(\alpha)\sin(\gamma)+\sin(\alpha)\sin(\beta)\cos(\gamma) & \sin(\alpha)\sin(\gamma)-\cos(\alpha)\sin(\beta)\cos(\gamma) & 0 \\ -\cos(\beta)\sin(\gamma) & \cos(\alpha)\cos(\gamma)-\sin(\alpha)\sin(\beta)\sin(\gamma) & \sin(\alpha)\cos(\gamma)+\cos(\alpha)\sin(\beta)\sin(\gamma) & 0 \\ \sin(\beta) & -\sin(\alpha)\cos(\beta) & \cos(\alpha)\cos(\beta) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$\begin{bmatrix} 1 & 0 & 0 & t_x \\ 0 & 1 & 0 & t_y \\ 0 & 0 & 1 & t_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix} \quad \textbf{(2.28)}$$

2.4.2.3. Addition of Scaling Parameters

Scaling provides zoom in / zoom out functionality for the registration operation. Registration of MRI images may require scaling of moving image because of the possible differences in data acquisition parameters. Thus, it is appropriate to plug scaling capability into the model, although it makes the model fail to be mathematically "*rigid*" as mentioned previously.

Scaling is handled by multiplication of the model given at the preceding subsection with a diagonal matrix, carrying scale coefficients at its diagonal:

$$\mathbf{T_{scaling2D}} = \begin{bmatrix} z_x & 0 & 0 \\ 0 & z_y & 0 \\ 0 & 0 & 1 \end{bmatrix}, \quad \textbf{(2.29)}$$

$$\mathbf{T_{scaling3D}} = \begin{bmatrix} z_x & 0 & 0 & 0 \\ 0 & z_y & 0 & 0 \\ 0 & 0 & z_z & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}. \quad \textbf{(2.30)}$$

Insertion of scaling matrices to the model raises the degree of freedom to 5 for 2D case and 9 for 3D case. Therefore, optimal parameter search requires

more computational effort and time compared to the rigid transformation case.

CHAPTER 3

IMPLEMENTATION

A software application has been developed in scope of this thesis study. Since it presents an agile programming environment with its built-in functions and compact toolboxes, MATLAB® is selected to be the development platform for the application.

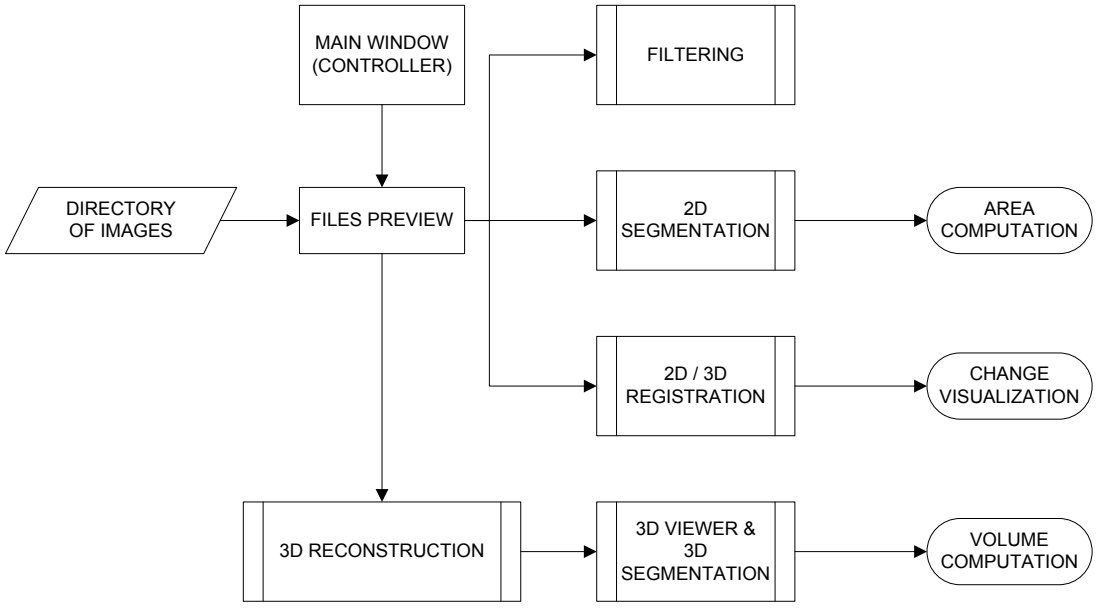

Figure 3.1 – General block diagram of the application

Figure 3.1 shows the overall structure of the application. Independent modules are controlled by a main controller, which enables the image directory or selected slice to be read and the image and the metadata information to be passed to the modules as inputs to be processed.

Image data, metadata information, and several other necessary parameters are passed together to the relevant module in a compact form using structured arrays. Application logic behind the computations and graphical user interfaces are programmed in separate files, in order to construct a layered architecture.

3.1. Filtering Module

Image filtering generally is sequentially the first group of the image processing operations. The group of operations can be classified as "*pre-processing*" in or scope, since they are not directly employed for analysis of images, but for preparing the images for further operations by performing some utilization such as elimination of noise and edge sharpening.

Following subsections explain explanation for three filters implemented within the application.

3.1.1. Linear Diffusion Filter

Recalling Equation **(2.2)**, $(C\Delta t/h^2)$ is renamed as ω - weighting factor - and the equation is modified as given below:

$$u_{i,j}^{k+1} = \omega\left(u_{i+1,j}^{k} + u_{i-1,j}^{k} + u_{i,j+1}^{k} + u_{i,-1j}^{k}\right) + (1 - 4\omega)u_{i,j}^{k}, \qquad \textbf{(3.1)}$$

with the constraint:

$$k = 1, \ldots, n; \qquad k \in \mathbb{Z}, n \in \mathbb{Z}, \qquad \textbf{(3.2)}$$

n is the positive integer iteration count, and $u_{i,j}^{k+1}$ expresses the gray level intensity of pixel (i, j) at iteration k. Therefore, filter has 3 inputs: image itself, weighting factor, and the number of iterations.

There exists two significant points in the implementation. First one is the stability condition over the weighting factor, which is given by Inequality **(2.3)**, the constraint over ω to be smaller than or equal to 2.5. In order to maintain this condition, user interface is programmed to allow inputs only inside the appropriate range. Second one is the programmatic application of "*Neumann Boundary Condition*", in order to preserve the mean value of the image, which is a common requirement arising for most of the modules. For this purpose, a helper function which reflects the boundary pixel frame to outside and creates a buffer region is generated for 2D and 3D images.

3.1.2. Perona-Malik Filter

Inserting Equation **(2.4)** into linear isotropic diffusion equation (Equation **(2.1)**) gives the Perona-Malik model - Equation **(3.3)**:

$$u_{i,j}^{k+1} = \left(\frac{g(|\nabla u|^2)\Delta t}{h^2}\right)\left(u_{i+1,j}^{k} + u_{i-1,j}^{k} + u_{i,j+1}^{k} + u_{i,-1j}^{k}\right)$$

$$+(1 - 4g(|\nabla u|^2)\Delta t/h^2)u_{i,j}^{k}. \quad \textbf{(3.3)}$$

Here, problem arises from the difficulty of expressing $u_{i,j}^{k+1}$ in explicit form, because, $|\nabla u|^2$ on the right-hand-side is the gradient value at the index (i, j) at the k^{th} iteration. An explicit scheme numerical solution of the model is given below:

$$u_{i,j}^{k+1} = \omega\left(C_E u_{i+1,j}^{k} + C_W u_{i-1,j}^{k} + C_N u_{i,j+1}^{k} + C_S u_{i,j-1}^{k}\right)$$

$$+ (1 - \omega(C_E + C_W + C_N + C_S))u_{i,j}^{k}. \quad \textbf{(3.4)}$$

In this equation, letter subscripts E, W, N, and S stands for the 4 directions, east, west, north, and south, and expresses directional gradients at the focused point. Equations from **(3.5)** to **(3.8)** give the idea in mathematical convention.

$$C_E = g(\left|u_{i-1,j}^{k} - u_{i,j}^{k}\right|), \quad \textbf{(3.5)}$$

$$C_W = g(\left|u_{i+1,j}^{k} - u_{i,j}^{k}\right|), \quad \textbf{(3.6)}$$

$$C_N = g(\left|u_{i,j+1}^{k} - u_{i,j}^{k}\right|), \quad \textbf{(3.7)}$$

$$C_S = g(|u^k_{i,j-1} - u^k_{i,j}|), \quad \textbf{(3.8)}$$

where, function g is given by Equation **(2.4)**.

Therefore, in addition to the input parameters of linear diffusion case, "*contrast threshold*" (λ) should be supplied to the filter manually.

3.1.3. Shock Filter

We start with the partial differential equation:

$$\partial u/\partial t = -sign(\nabla^2 u) \times |\nabla u|. \quad \textbf{(3.9)}$$

"*Shock Filter*" is based upon the principle adaptive backward / forward differencing, namely, "*upwind derivatives*". If the sign of the Laplacian term is positive:

$$\partial u/\partial t = -|\nabla u|, \quad \textbf{(3.10)}$$

filter applies erosion around minima, and if the sign of Laplacian term is negative:

$$\partial u/\partial t = |\nabla u|, \quad \textbf{(3.11)}$$

filter applies dilation around maxima. With these operations filter has an effect of sharpening on input image. Numerical expressions for dilation and erosion operations are separate and Equations from **(3.12)** to **(3.16)** give these numerical expressions:

$$|\nabla u| = \sqrt[2]{({u_x}^2 + {u_y}^2)} \qquad \textbf{(3.12)}$$

where;

$${u_x}^2 = \{\min((u_{i,j} - u_{i-1,j})/h_x,\ 0)\}^2 + \{\max((u_{i+1,j} - u_{i,j})/h_x,\ 0)\}^2, \qquad \textbf{(3.13)}$$

$${u_y}^2 = \{\min((u_{i,j} - u_{i,j-1})/h_y,\ 0)\}^2 + \{\max((u_{i,j+1} - u_{i,j})/h_y,\ 0)\}^2 \quad . \qquad \textbf{(3.14)}$$

Equations **(3.13)** and **(3.14)** applies if the sign of Laplacian term is negative (operation is dilation). If the operation is erosion, equations **(3.15)** and **(3.16)** is applied:

$$u_x{}^2 = \{\max((u_{i,j} - u_{i-1,j})/h_x\,,\ 0)\}^2 + \{\min((u_{i+1,j} - u_{i,j})/h_x\,,\ 0)\}^2,$$

(3.15)

$$u_y{}^2 = \{\max((u_{i,j} - u_{i,j-1})/h_y\,,\ 0)\}^2 + \{\min((u_{i,j+1} - u_{i,j})/h_y\,,\ 0)\}^2.$$

(3.16)

Discrete solution for the Laplacian term is given in Equation **(3.17)**.

$$\nabla^2 u = \{u_{i+1,j} + u_{i-1,j} + u_{i,j+1} + u_{i,j-1} - 4 \times u_{i,j}\}/h^2. \qquad \textbf{(3.17)}$$

If we combine all of the equations above, and rewrite left-hand-side as:

$$\partial u/\partial t = (u_{i,j}^{k+1} - u_{i,j}^{k})/\Delta t. \qquad \textbf{(3.18)}$$

We have an explicit scheme discrete solution for the model of the "*Shock Filter*" given by **(3.9)**. The only trick in the implementation of the code is programming mentioned selective behavior with the computation of the upwind derivatives ($|\nabla u|$).

3.2. Image Segmentation Module

Image segmentation module is implemented based on Ambrosio-Tortorelli minimization model for the Mumford-Shah energy functional for the reasons given in Chapter 2. Derivation of the numerical solution to the Ambrosio-Tortorelli minimization starting from Mumford-Shah energy functional is given in Appendix A.

3.2.1. 2D Image Segmentation

Input parameters for the Ambrosio-Tortorelli segmentation model are:

- Image itself,
- Regularization coefficient,
- Data fidelity factor,
- Edge complexity term,
- Maximum number of the iterations, and,
- Saturation tolerance.

“Regularization coefficient”, “data fidelity factor”, and “edge complexity term” are the name representations for parameters, α, β, and ρ of Equation **(A.1)**, respectively.

As shown on Figure 3.2, the first operation to do is to compute the starting values for signals u and v. Since u represents segmented image, it is appropriate to choose u_0 as the input image z. A proper starting value for v is

given by Equation **(3.19)**. It puts Euler-Lagrange representation of v (Equation **(A.4)**) into the form of "*Heat Equation*".

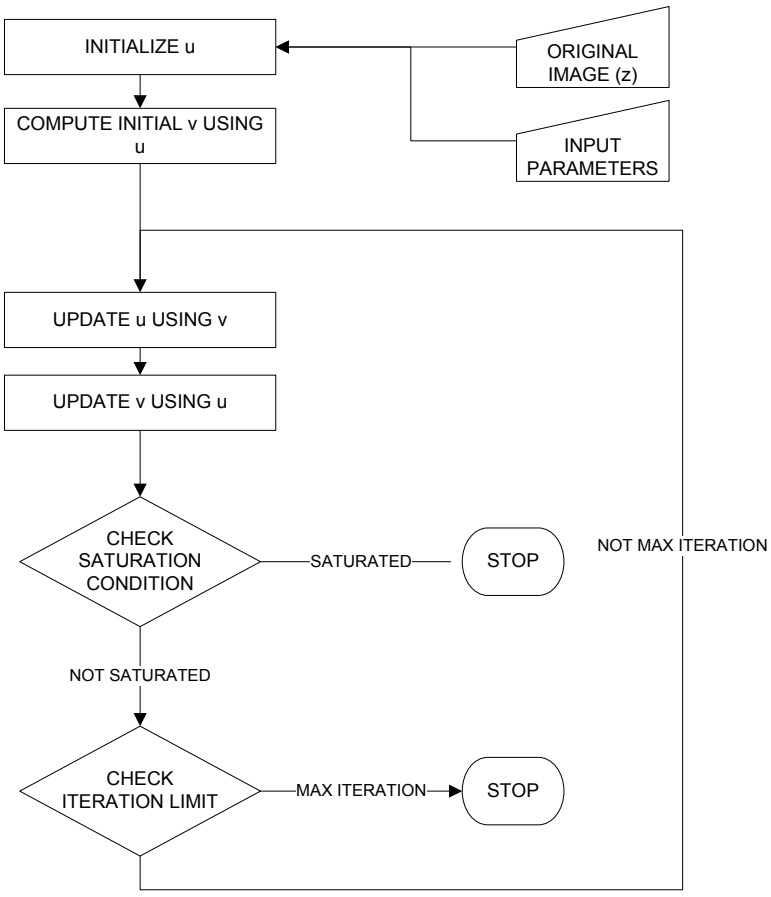

Figure 3.2 – Basic flowchart for the implementation of Ambrosio-Tortorelli segmentation

$$v_0 = 2\alpha|\nabla u|^2/(2\alpha|\nabla u|^2 + 1) \qquad \textbf{(3.19)}$$

In the main loop, u and v values are updated sequentially, according to the rules defined by Equations **(A.5)** and **(A.9)** using input parameters z, α, β, and ρ.

There exist two criteria to determine whether the computation loop ends or continues. First one is the saturation condition defined by Equation **(3.20)**. Second is obviously reaching to the maximum number for iterations. As far as none of the stopping conditions is not met, application continues to run for updating the functions u and v:

$$|u^{t+1} - u^t| < \epsilon |u^t|. \qquad \textbf{(3.20)}$$

As shown by the above equation, saturation - in this context - means approach of rate of change of a total measure over image intensity to zero. It is appropriate to select ϵ (saturation tolerance) relatively small compared to average image gray level, for example 10^{-6} or 10^{-7}.

3.2.2. Feature Area Computation

For the purpose of computing cross-sectional area of an image feature using a 2D image, a compositional methodology consisting Mumford-Shah segmentation and region growing algorithm is developed. Mumford-Shah segmentation procedure is applied as explained above, and it gives the output image when saturation condition is met or maximum allowed number for the iterations is reached. Additionally, iterations can be stopped manually by the user, who can view the instantaneous state of the evolving output during the procedure. As a result, an output composed of smoothed regions is formed.

Region growing algorithm is employed for the purpose of selection of the desired feature, and creation of a binary form of the output. Neighborhood radius is constant and selected as 1 pixel. Threshold range - lying between 0

and 1 - is defined by the user. Resulting binary image carries the information of the positions for pixels included by selected object (or region).

Metadata of any DICOM file is read into the memory during the file selection state. Built-in function "*dicominfo*" of MATLAB® Image Processing Toolbox is used to acquire DICOM metadata into a structured array containing DICOM data fields as its cells.

In case of MRI images, existence of a field named "*PixelSpacing*" is obligatory in DICOM metadata. Field stores an *n*-dimensional vector carrying double values, which gives the information of real pixel size for each dimension as millimeters. Using binary image formed by region growing and pixel spacing information, desired cross-sectional area can be computed as pixels and millimeter-squares.

A study on validation of the process is given on Chapter 4, and results with figures containing screenshots and relevant data are presented in Chapter 5.

3.3. Image Registration Module

As mentioned before, developed application contains an independent module for the purpose of registration of two images. Next subsection introduces implementation of registration of 2D images with manually acquired known correspondence. Following subsection gives the details in implementation of the fully-automated 2D and 3D image registration process.

3.3.1. Registration with Known Correspondence

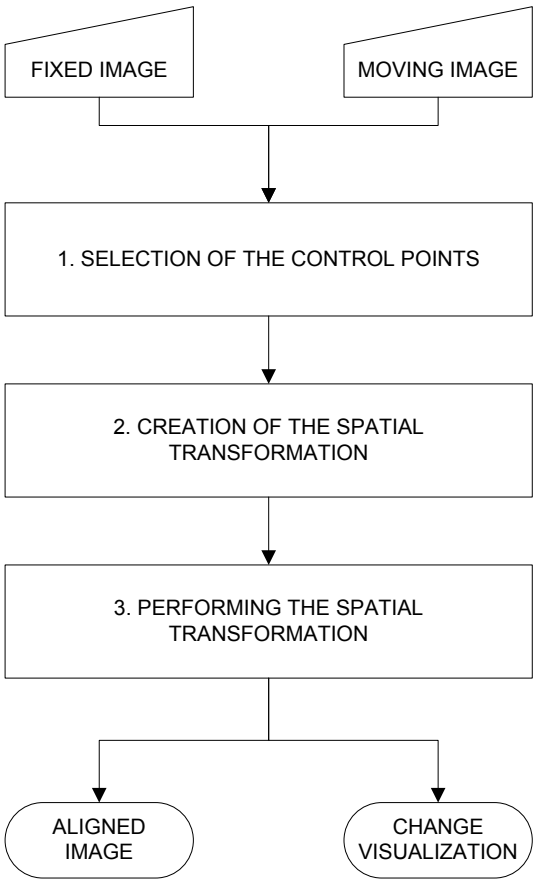

Figure 3.3 – Flowchart for the implementation of registration with known correspondence

As seen from Figure 3.3, image registration with known correspondence involves three consecutive steps after acquisition of 2D images. These three steps mainly implemented using proper built-in function for each one.

First requirement for the procedure is the selection of the control points. MATLAB® Image Processing Toolbox function "*cpselect*" provides a user interface customized for interactive point selection, and returns the spatial coordinate pairs for sequentially selected correspondence points from the image pair. Second, the spatial transformation model is prepared using "*cp2tform*". This function takes control point pairs, and the desired

transformation model as input arguments, and returns the *TFORM* structure ready to use as an input parameter for the built-in function "*imtransform*", which handles the transformation operation. Minimum number of control point pairs to be selected by user varies depending on the transformation type.

Table 3.1 – Transformation types for cp2tform function [69**]**

TRANSFORMATION TYPE	DESCRIPTION	MINIMUM NUMBER OF CONTROL POINTS
Non-reflective similarity	Combination of translation, rotation, and scaling.	2 Pairs
Similarity	Same as "non-reflective similarity" but with the addition of optional reflection.	3 Pairs
Affine	Straight lines remain straight, parallel lines remain parallel. Shear functionality is added to "similarity" transformation.	3 Pairs
Projective	Straight lines remain straight, parallel lines converge to a vanishing point inside or outside image domain.	4 Pairs
Polynomial	Used when the image are curved. Higher order gives better results.	6 Pairs (Order 2) 10 Pairs (Order 3) 15 Pairs (Order 4)
Piecewise-linear	Used when different regions of the image looks distorted differently.	4 Pairs
Local weighted mean	Used when distortion varies locally and "piecewise-linear" is not sufficient.	6 Pairs (12 Recommended)

Table 3.1 [**69**] gives the valid transformation types, brief descriptions for them, and the minimum required number of control point pairs for each type of transformation. Type of the transformation is selected interactively by the

user with given constraint over the number of control point pairs. *TFORM* transformation structure is prepared and applied on the moving image.

Visualization of change in image is obtained by computing the absolute difference between the fixed image and the aligned image.

3.3.2. Fully-Automated Global Image Registration with Rigid-Body Transformation

Figure 3.4 shows general flow of the steps involved in the automated image registration process.

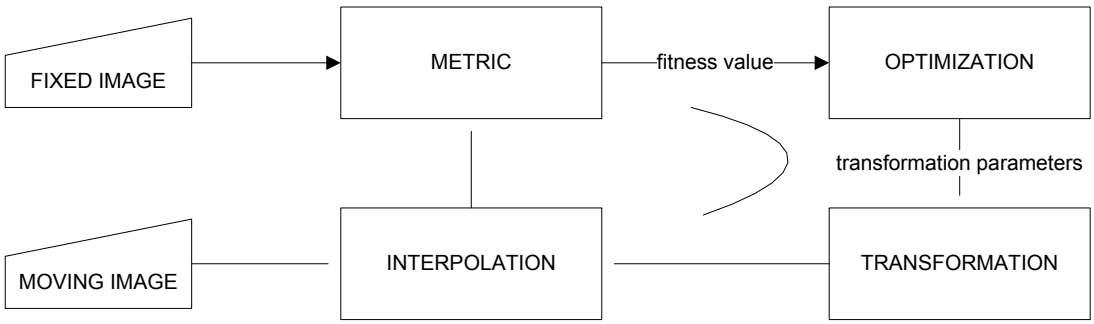

Figure 3.4 – Fully-automated registration flowchart

3.3.2.1. Interpolation

Interpolation is the process of defining a function that takes on specified values at specified points. In digital signal processing, interpolation can be defined as the process of estimation of values at the unknown data points (non-grid positions) by use of known samples. Moving image is resampled in order to be synchronized with the fixed image in terms of dimensions.

Various methods can be employed for this purpose, such as, "*sinc interpolation*", "*nearest neighbor interpolation*", "*linear interpolation*", "*quadratic interpolation*", "*B-spline interpolation*", "*cubic interpolation*", "*Lagrange interpolation*", and "*Gaussian interpolation*" [**61**,**70**]. In scope of our study, "*linear interpolation*" method is used.

3.3.2.2. Similarity Metric

Metric is the mathematical similarity measure between two functions. In other words, metric compares how well the two images match each other. Resampled moving image and the reference (fixed) image are compared with regard to the predetermined metric. Intensity-based relations like "mean squares", "normalized correlation", and "mean reciprocal squared difference"; information theoretic relations like "mutual information" and "joint entropy"; and histogram-based methods like "joint histogram estimation" can be used as similarity metric between the image functions.

Iterations stop when the calculated metric value is equal to or smaller than the determined error tolerance; therefore, selected metric decides if the instantaneous transformation model provides correctness at sufficient level for the registration result.

In this work, "*normalized cross-correlation*" is used. Mathematical expression for this metric is given on Subsection 4.2.1.

3.3.2.3. Optimization

Calculated similarity measure (metric) is compared with the constant error tolerance and if the necessary condition is not met, transformation model is

applied to the moving image again. Before the reapplication of transformation model, input parameters of the model must be recalculated using the feedback from the previous iteration. The component which handles this recalculation process and updates the input parameters is called optimizer.

For the optimization of transform parameters, many well-known "*gradient-descent*" or "*stochastic-based*" optimization algorithms can be used. Selected optimization algorithm is the leading factor determining success and computational time of the registration process.

In this thesis study, a built-in function of MATLAB® Optimization Toolbox called "*fminsearch*" is used in order to minimize the minus of normalized cross correlation. Selected method is an implementation of Nelder-Mead Simplex Algorithm, which is first published in 1965 [**71**]. Details involving use of and convergence criteria over this optimization method are given on reference [**72**]. Required additional inputs such as moving image data and the bounds for optimization parameters are injected into the optimization algorithm using "*anonymous function declaration*" capability of MATLAB®.

3.3.2.4. Transformation

Final step of the iteration cycle is the application of transformation model on the moving image. This part can simply be defined as the multiplication of the moving image with the constructed transformation matrix in order to obtain the registered image. Selected transformation model defines the constraints on the alterations on moving image during image registration process. In other words, scope of the changes on moving image is determined by the chosen transformation model.

As the result of iterations, correct transform input parameters are calculated. The transformation model constructed by use of aforesaid parameters is

applied onto the original version of the moving image, and the registered image is acquired.

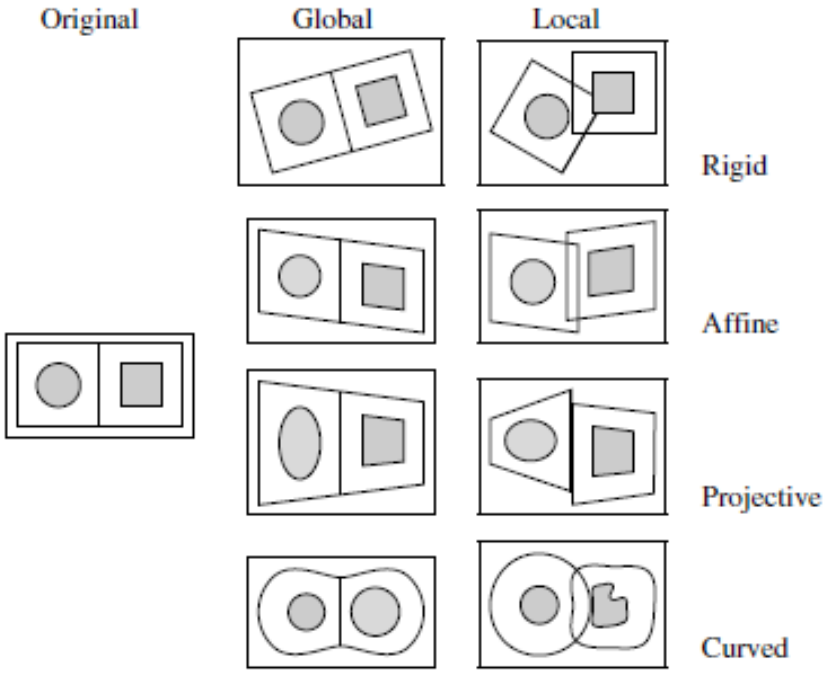

Figure 3.5 – Visualization of transformation types, taken from reference [61**]**

Figure 3.5 is taken from the article given in reference [**61**], and is given in order to represent visualizations for different transformation types. As mentioned in Subsection 2.4.2, we have implemented 2D and 3D global rigid transformation in scope of this thesis work.

Table 3.2 – Number of transformation parameters to be optimized

	TRANSLATION	**ROTATION**	**SCALING**	**TOTAL (max)**
2D	2	1	2	**5**
3D	3	3	3	**9**

Table 3.2 shows the maximum number of transformation parameters (degree of freedom) to be optimized. As mentioned before, high number of parameters brings more computational load, therefore results in more divergence risk and more computational time.

Search for scaling parameters are optional. If they strictly rigid registration is selected, their value is kept as 1.

3.4. 3D Medical Image Viewer and 3D Image Segmentation

A 3D image viewer is implemented as a separate module in order for system users to be able to browse volumetric images, focus on desired volumetric region, select rectangular region of interest, and apply implemented operations over the targeted region.

3.4.1. 3D Medical Image Viewer

Selected set of 2D slices is used for reconstruction of volumetric image. User manually selects the image directory, and the orientation (*axial*, *sagittal*, or *coronal*) of the volumetric data. During the reconstruction process, pixel spacing and slice thickness are read form the DICOM metadata, and ratio between them is considered for interpolation in order to view the image in real proportional sense. Interpolation is also applied for memory optimization. The maximum allowed volumetric image size is pre-defined (arranged properly for 32-bit memory) and bigger size images are reconstructed with resizing.

3D image viewer is utilized with capability of focusing on a user-specified point. This can be done simply by clicking on the image window, or browsing the image in three projections using the horizontal and vertical scrollbars.

Selection of a 3D rectangular region of interest is also made possible within the viewer. 3D segmentation and visualization of 3D model are applied only within the selected region of interest considering the performance and memory issues.

Additionally, thresholding capability is implemented on the screen in order for user to be able to eliminate irrelevant texture and focus on the desired region quickly and easily.

Figure 3.6 shows an example use of the 3D medical image viewer module in the thresholding mode.

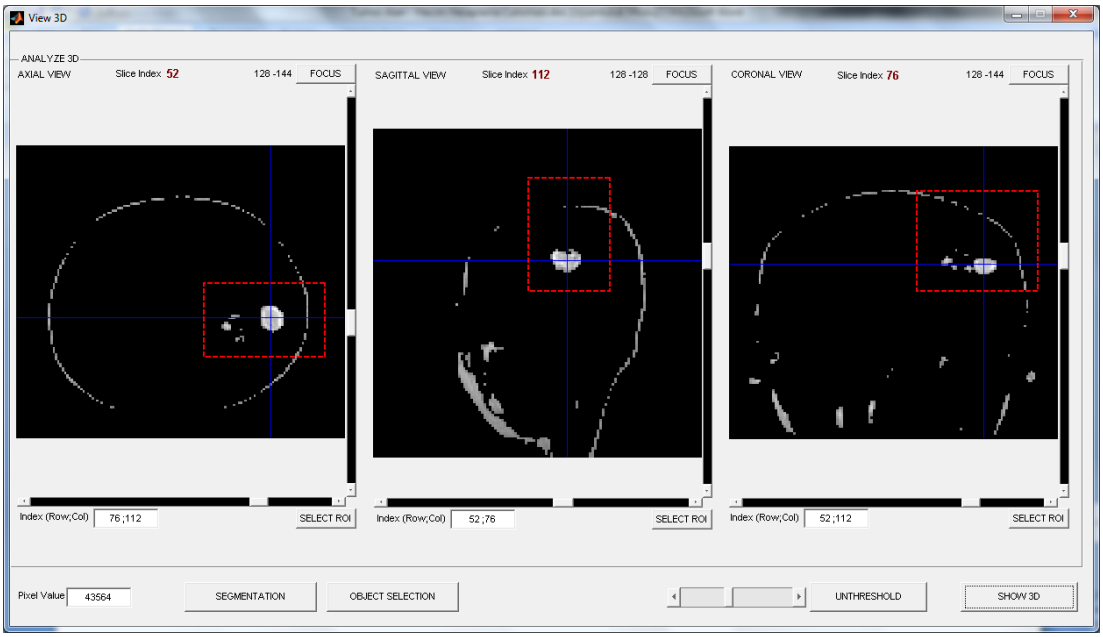

Figure 3.6 – A screenshot of the 3D medical image viewer module in thresholding mode.

3.4.2. 3D Segmentation

Ambrosio-Tortorelli minimization method for Mumford-Shah segmentation is modified in order to handle segmentation of 3 dimensional data. Representing the third dimension, z is inserted into Equations from **(A.1)** to **(A.10)**, and equations are properly redefined.

Interpolation procedure mentioned in previous subsection is critical for 3D segmentation, because interactions between neighbor voxels are kept independent from the direction of neighborhood in modification of the equations for 3D case.

Application of 3D segmentation creates a volumetric output image inside the boundaries of region of interest, which is formed of relatively smooth 3d regions. These regions can be modeled and viewed as 3D together with selection of proper edge and object thresholds, as well as they can be used for volume computation separately. Examples of results are presented in Chapter 5.

3.4.3. Feature Volume Computation

Similar to the cross-sectional area computation case (Subsection 3.2.2), a sequential application of Mumford-Shah segmentation and region growing algorithm is utilized in order to be able to quantify volume of selected object. After segmentation of the region of interest, a modified 3D version of the region growing algorithm is used in order to select the target object. Object is selected over a binary representation of the segmented image, and volume is calculated using dimensional data in the structured array carrying DICOM file metadata.

In addition to the "*PixelSpacing*" vector, "*SliceThickness*" field of the metadata is used as the third dimension. If the MRI image is not acquired as volumetric, value for the field "*SpacingBetweenSlices*" is different from zero, and mostly greater than the slice thickness. In this case, value of this field is used as the 3rd dimension.

As the result of operations, volume of the selected object is given as voxels and millimeter cubes. Sample experiments and results are presented on Chapter 5.

CHAPTER 4

PERFORMANCE EVALUATION

Performances of two main modules - image segmentation and image registration - of the application are evaluated in scope of this thesis study. Validation methods and results are given in following sections of this chapter.

4.1. Validation of Image Segmentation Module

4.1.1. Metrics

4.1.1.1. Mean Preservation

As far as "*Neumann Boundary Condition*" is satisfied, there exists no sinks or sources in the image domain with evolving operations. Therefore, average intensity value for pixels remains constant. In other words, operation is mean preserving:

$$\partial y/\partial n\,(x) = \nabla y(x) \cdot \boldsymbol{n}(x). \qquad \textbf{(4.1)}$$

In Equation **(4.1)**, $\boldsymbol{n}$ denotes the normal to the boundary. In order for the dot product to be equal to zero, gradient of y in the direction of normal vector should be equal to zero. As explained in Chapter 3, boundary pixels are reflected along the outside frame of the image with a helper function.

Mean preservation condition is checked during the experiments, in order to make sure there has not been any mistakes in the implementation phase related to existing sources or sinks on boundaries. Results are presented in Subsection 4.1.2

4.1.1.2. Performance Measures

A database of segmented and non-segmented versions of brain MRI images which form ground truth information for validation, and the metrics for the evaluation of success in image segmentation are presented in references [**73**,**74**] by Laboratory of Neuro Imaging at UCLA (University of California, Los Angeles). In this thesis work, six of given performance measures (metrics) are used for evaluation of the segmentation process, which are explained in detail in this subsection.

Each one of these metrics is expressed in terms of regions, formed within the interaction of a "*truth set*" and a "*results set*", elements of which are pixel positions inside target region over ground truth, and pixel positions inside target region over experimental result. As shown on Figure 4.1 [**74**], domain is separated into 4 sub-domains, namely, "*true negative* ($\boldsymbol{TN}$)", "*true positive* ($\boldsymbol{TP}$)", "*false negative* ($\boldsymbol{FN}$)", and "*false positive* ($\boldsymbol{FP}$)".

"*Jaccard similarity index*" (also known as "*Tanimoto coefficient*") is defined as the size of the intersection divided by the size of the union. Equivalently:

$$J(A,B) = |A \cap B|/|A \cup B|, \qquad \textbf{(4.2)}$$

for sets A and B. Value of J is ideally 1 for an ideal segmentation of the desired region. In our case, J can be expressed as:

$$J = \boldsymbol{TP}/(\boldsymbol{FP} + \boldsymbol{TP} + \boldsymbol{FN}). \qquad \textbf{(4.3)}$$

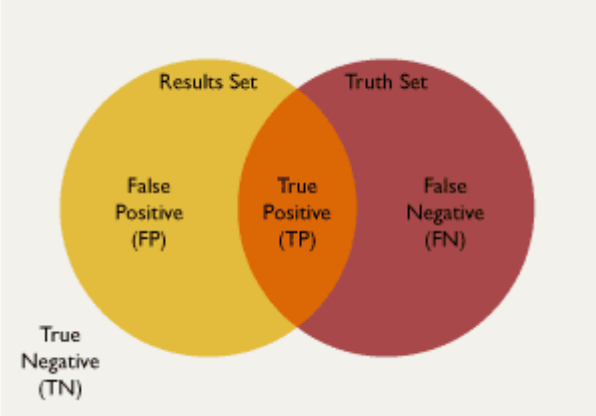

Figure 4.1 – Domain of metrics used for validation of segmentation, taken from reference [74]

Another metric measuring "*set agreement*" is called as "*Dice coefficient*", and mathematical representation of this measure is given by Equation **(4.4)**. Equation **(4.5)** shows formulation in terms of our domain:

$$D(A,B) = 2|A \cap B|/(|A| + |B|), \qquad \textbf{(4.4)}$$

$$D = 2\boldsymbol{TP}/(\boldsymbol{FP} + 2\boldsymbol{TP} + \boldsymbol{FN}). \quad \textbf{(4.5)}$$

Obviously, ideal value for "*Dice coefficient*" is 1, as it is for the "*Jaccard similarity*". For both of the cases, range of the value is between 0 and 1. Equations from **(4.6)** to **(4.9)** show other metrics used for performance evaluation of segmentation:

$$sensitivity = |\boldsymbol{TP}|/|\boldsymbol{TP} + \boldsymbol{FN}|, \quad \textbf{(4.6)}$$

$$specificity = |\boldsymbol{TN}|/|\boldsymbol{TN} + \boldsymbol{FP}|, \quad \textbf{(4.7)}$$

$$false\ negative\ rate = |\boldsymbol{FN}|/|\boldsymbol{FN} + \boldsymbol{TP}| = 1 - sensitivity, \quad \textbf{(4.8)}$$

$$false\ positive\ rate = |\boldsymbol{FP}|/|\boldsymbol{TN} + \boldsymbol{FP}| = 1 - specificity. \quad \textbf{(4.9)}$$

As it can be easily observed form the equations, ideal value for $sensitivity$ and $specificity$ is again 1, and ideal value for $false\ negative\ rate$ and $false\ positive\ rate$ is 0.

4.1.2. Results

Performance evaluation experiments are made on two classes of images which are:

- Original and deformed versions of simulated images including arbitrary shaped geometrical data, and
- Original and segmented versions of real MRI data.

Figure 4.2 – Original simulated image, taken from reference [74]

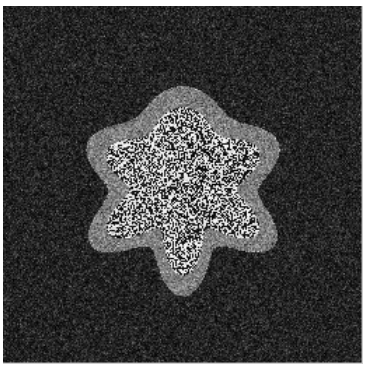

Figure 4.3 – Deformed simulated image (lower SNR), taken from reference [74]

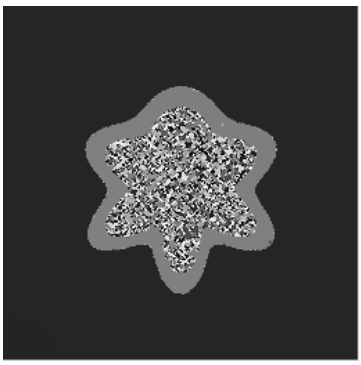

Figure 4.4 – Segmented image

Figure 4.5 – Selected region

These samples are provided by reference [**74**]. Figure 4.2 and Figure 4.3 shows original and deformed versions of the simulated image, which is composed of three uniform regions. Performance is evaluated in terms of

mean preservation and the metrics defined above, by the segmentation of gray region separating the black background and the white region located at the center.

Table 4.1 – Performance evaluation results for simulated data

Jaccard	Dice	Sensitivity	Specificity	FNR	FPR
0.908200	0.951890	0.984190	0.992430	0.015815	0.007571

Figure 4.4 shows the output of segmentation operation applied over complete image domain. Inputs are given as "*regularization factor*=10", "*data fidelity coefficient*=100", and, "*edge complexity factor*=0.035". Result is taken at 5000 iterations. *Mean value* of pixel intensities is observed to be *constant* over time and is equal to 0.2221, which is same with the original image.

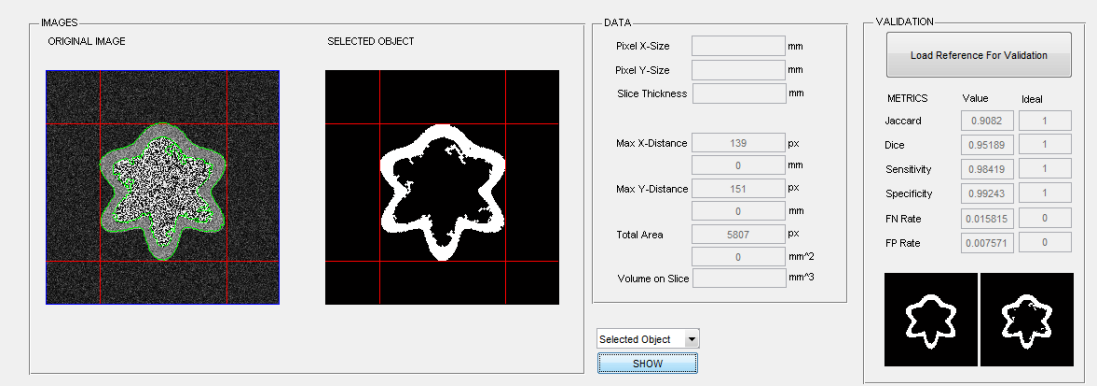

Figure 4.6 – User interface for area representation and performance evaluation

Table 4.1 presents the performance evaluation metric values for the experiment done by the simulated data. As seen from the table, the first four measures, which are ideally equal to 1 are all above 0.9 (error < 10%), and "*false negative rate (FNR)*" and "*false positive rate (FPR)*" metrics are both close to 1%.

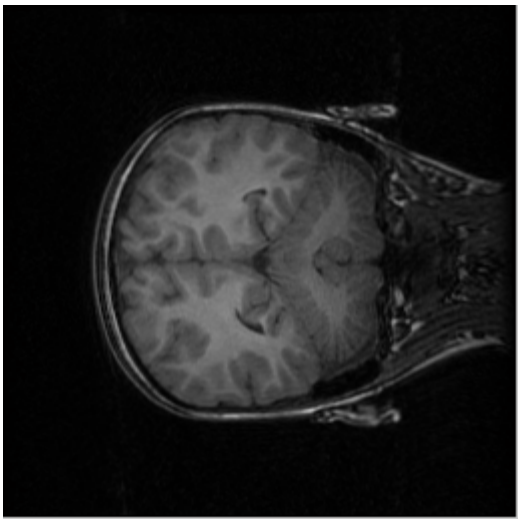

Figure 4.7 – Original MRI image, taken from reference [74]

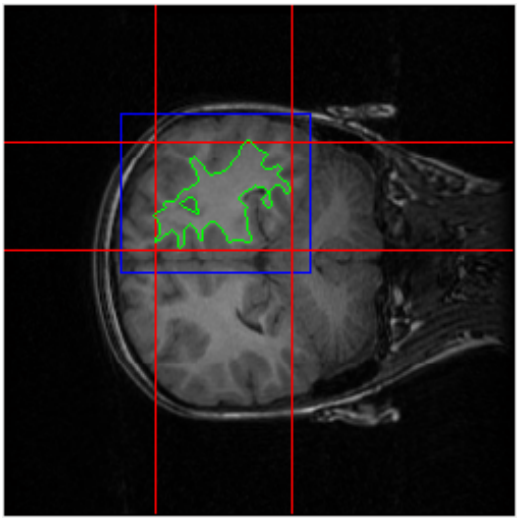

Figure 4.8 – Segmented MRI image

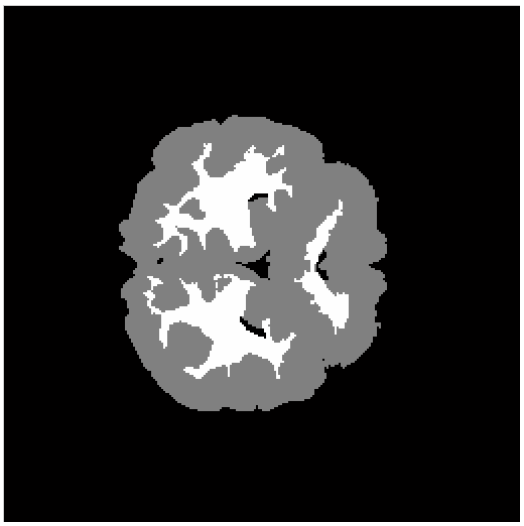

Figure 4.9 – Ground truth segmentation, taken from reference [74]

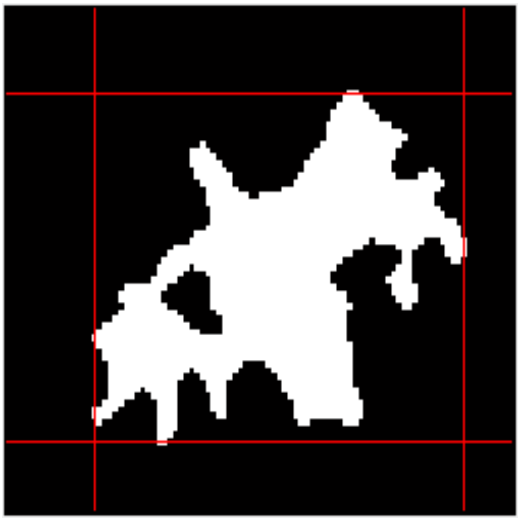

Figure 4.10 – Selected Region

The second experiment is implemented with the application of segmentation procedure on real T1-MRI brain data belonging to a male subject, which is shown on Figure 4.7. Blue rectangle over Figure 4.8 shows selected region of interest (ROI). Inputs are given as "*regularization factor*=10", "*data fidelity coefficient*=100", and, "*edge complexity factor*=0.035". Result is taken at 5000 *iterations* performed in 82 *seconds*. *Mean value* of pixel intensities is observed to be *constant* over time and is equal to 0.4735, which is same with the original image over selected ROI.

Table 4.2 – Performance evaluation results for real MRI data

Jaccard	Dice	Sensitivity	Specificity	FNR	FPR
0.851810	0.919970	0.95418	0.969430	0.045817	0.030574

Table 4.2 introduces the numerical results for the measures of performance evaluation. As shown on the table, values for "*Dice coefficient*", "*Sensitivity*", and "*Specificity*" are above 0.9 and *FNR* and *FPR* rates are below 10%.

4.2. Validation of Image Registration Module

Performance of automated image registration is evaluated using an original image as "*fixed image*" and a transformed image (with known parameters) as "*moving image*". Image visuals, parameters, and results are presented in Subsection 4.2.2.

4.2.1. Normalized Cross-Correlation Metric

Value of the similarity measure between the aligned version of the moving image and the fixed image constitutes a stopping condition for the cycle of iterations. Thus, this value of the metric is also a measure of success of the operation.

Mathematical representation of normalized cross-correlation between signals f and g is given below:

$$\frac{1}{n-1}\sum_{x,y}\frac{(f(x,y)-\bar{f})(g(x,y)-\bar{g})}{\sigma_f \sigma_g}. \qquad \textbf{(4.10)}$$

In Equation **(4.10)**, $\nexists$ represents mean value of f, and σ_f represents standard deviation of f. n is the total number of pixels.

Normalized cross-correlation is numerically implemented for registration of 2D and 3D images.

4.2.2. Results

Figure 4.11 – Fixed image

Figure 4.12 – Moving image

Original and distorted images used as fixed and moving images are shown on Figure 4.11 and Figure 4.12, respectively. Rotation, translation, and scaling is applied on the fixed image to reproduce the moving image.

First row of Table 4.3 shows the values of parameters in order to induce transformation to the original image. Expected correction values for "*rotation*" and "*translation*" parameters are additive inverses of the induced values, and expected correction values for "*scaling (zoom)*" parameters are multiplicative inverses of the induced values. As seen on the table, results are reasonable.

Table 4.3 – Transformation parameters

	Rotation (degrees)	X-translation (pixels)	Y-translation (pixels)	X-zoom (rate)	Y-zoom (rate)
Induced	-3.5	-20	1	1.1	1.1
Initials	-4	10	-10	1	1
Corrected	3.558	20.7972	-1.1207	0.9107	0.90049

Figure 4.13 – Aligned image

Figure 4.14 – Absolute difference of aligned and fixed images

Normalized cross-correlation value between the fixed and aligned image is computed as 0.97866, as a result of fully-automated registration operation. Process is completed in 186 iterations in 95 seconds, with arbitrarily chosen initial values given in Table 4.3. Figure 4.13 and Figure 4.14 shows resulting (aligned) image and the absolute difference between aligned image and fixed image.

CHAPTER 5

EXPERIMENTAL RESULTS

Several experiments related to each module has been done and results are presented and discussed in this chapter. Related visual demonstrations for user interfaces, inputs and outputs are presented with numerical results and interpretations for each section.

5.1. Image Smoothing and Restoration

Original brain MRI image given on Figure 5.1 is intentionally degraded with different types of noise in order to construct noisy data for filtering experiments. Image on Figure 5.2 is created with inserting 2% "*salt & pepper noise*" into the original image. Similarly, the image given as Figure 5.3 is built with insertion of "*Gaussian white noise*" into the original image with zero *mean* and 0.01 *variance*. Final image (Figure 5.4) shows a combination of two mentioned types of noises.

Experiments involve application of filters onto these images and interpretation of the results.

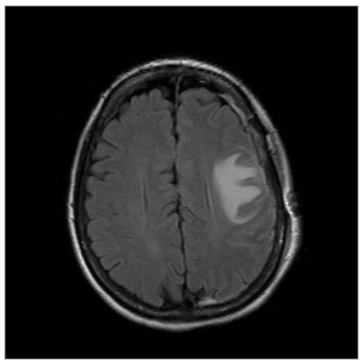

Figure 5.1 – Original image, taken from reference [75]

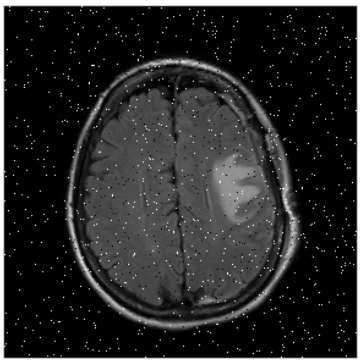

Figure 5.2 – Image with salt & pepper noise

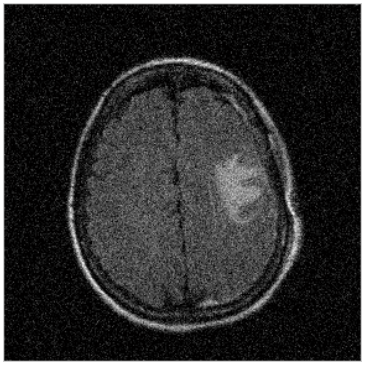

Figure 5.3 – Image with Gaussian white noise

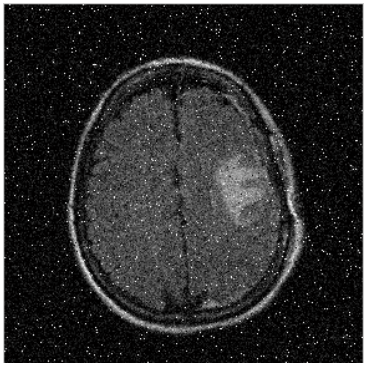

Figure 5.4 – Image with both salt & pepper and Gaussian white noises

5.1.1. Linear Diffusion Filter

Linear diffusion filter (LDF) is basically used for elimination of unwanted noise from the image. Main drawback of linear diffusion operation is causing loss of meaningful information due to blurring effect.

Figures Figure 5.6 and Figure 5.7 are acquired by linear diffusion filter applied with 100 iterations and with a weighting factor of 0.2 on the image with salt & pepper noise. As seen on the second figure, noise is completely eliminated. However, in case of medical image processing, operation does not give a meaningful result. As it can be observed through the image on the right, absolute difference between input and output has high values along high gradient regions; in other words, edges. Therefore, it can be said that linear diffusion causes loss of edge information over images.

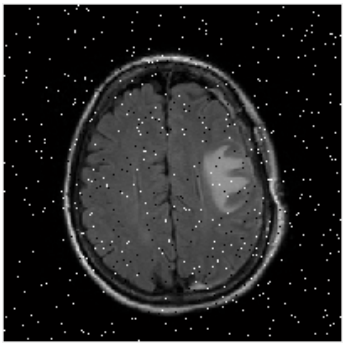

Figure 5.5 – Input image with salt & pepper noise

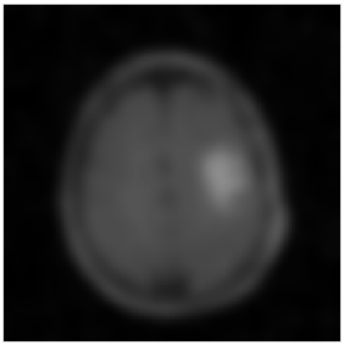

Figure 5.6 – Linear diffusion output (100 iterations)

Figure 5.7 – Absolute difference between input and output

Figure 5.8 – Linear diffusion output (10000 iterations)

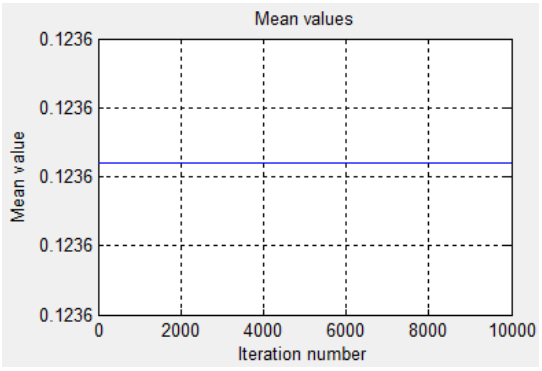

Figure 5.9 – Mean value vs. iterations (LDF)

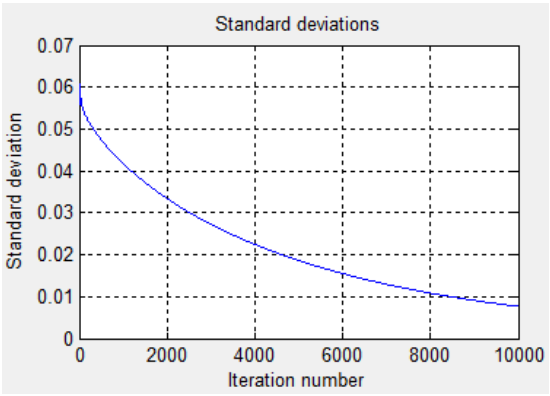

Figure 5.10 – Standard deviation vs. iterations (LDF)

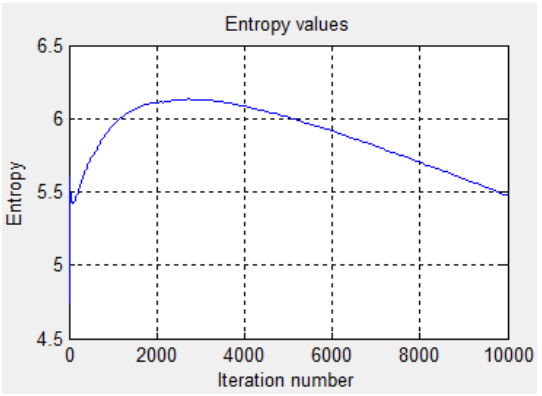

Figure 5.11 – Entropy vs. iterations (LDF)

Figures from Figure 5.8 to Figure 5.11 show output and statistical evaluation yielding linear diffusion with 10000 iterations and weighting factor of 0.2. Output image is almost a uniform gray image as expected. Constant mean

value over time shows correct implementation of "*Neumann boundary condition*".

Standard deviation has changed with a pattern of a rapid increase followed by exponential decay. The increase can be explained by amplification of impulsive pixels caused by salt & pepper noise at early iterations.

Entropy is a measure of amount of information carried by a signal. Looking at Figure 5.11, similar to the standard deviation, iterations up to some instant cause increase in the entropy. Afterwards, decay occurs in the entropy value. It is reasonable since entropy of a uniform image is equal to zero.

Finally, "*point spread function (PSF)*" of LDF is visualized with Figure 5.12. Relation between an iterative LDF and Gaussian convolution can easily be observed from the figure.

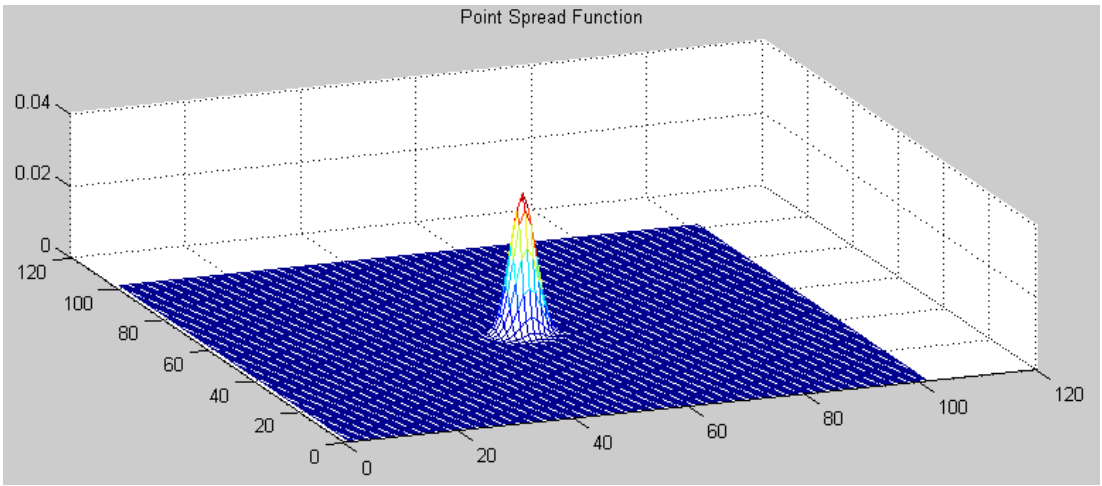

Figure 5.12 – Point spread function (PSF) for LDF

PSF is created by applying LDF over a 101 x 101 image which is zero everywhere but at the central pixel. Pixel at position (51, 51) carries a value of 1. Mathematical relation between "*Gaussian convolution*" and "*linear diffusion filtering*" is deeply investigated in reference [**8**].

5.1.2. Perona-Malik Filter

Perona-Malik filter requires three inputs from the user. First one is the *weighting factor*, which affects the process in the same way as it does in linear diffusion case. Rate of regularization is proportional with the magnitude of the *weighting factor*, which can have values between zero and 0.25 in order not to violate stability constraints. Second input is the number of iterations. Filtering operation has no other stopping criterion than the total number of iterations. Iteration count symbolizes time parameter in PDEs; so, it is expected for image to evolve into a uniform gray image after sufficient number of iterations applied. Third input parameter is the *contrast threshold*, which creates a control over diffusion rate depending on gradient at relevant position. Increasing the *contrast threshold* increases diffusion rate, consequently causes loss of edge information. Therefore, it is appropriate to choose a relatively small value (in the order of 0.001) for *contrast threshold*.

Perona-Malik filter (non-linear isotropic filter) can be evaluated as a primitive form of manual segmentation, because of its edge preserving property. Of course this discourse is valid up to some degree, because as time goes to infinity, it is mathematically guaranteed for the filter to convert to image into a single gray partition.

Perona-Malik filter is expected to work well with *Gaussian white noise*; however, it fails in case of *salt & pepper noise* because of the characteristics of that kind of degradation. In order to be able to observe the effects of the filter on these two deformation type simultaneously, image given by Figure 0.4 (Image with both Gaussian whit noise and salt & pepper noise) is used as input for the experiments involved in this subsection.

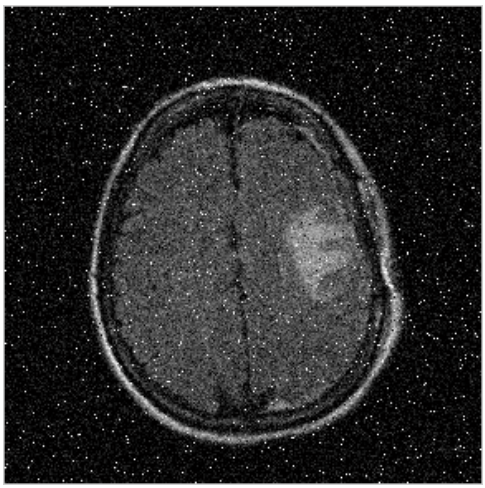

Figure 5.13 – Input image deformed by both Gaussian white noise and salt & pepper noise

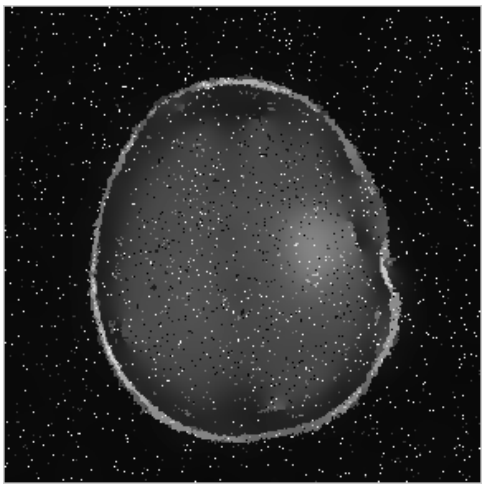

Figure 5.14 – Non-linear isotropic filter output with contrast threshold = 0.005

Output image given by Figure 5.14 is acquired by applying Perona-Malik on the image on the left-hand-side, with a *weighting factor* of 0.25, and *contrast threshold* of 0.005, at 200 iterations. Black background is observed to be regularized (white noise is eliminated) except impulsive pixels (salt & pepper component of the input image). Outer boundaries of the skull have been protected from diffusion; however, inner head has been regularized as if it is a single object. Edge information related to the tumor region and brain boundaries has been completely lost with that particular parameter set.

In order to preserve the edges of edema region while preserving the regularization rate, magnitude of *contrast threshold* should be decreased.

Figure 5.15 is produced with *contrast threshold* value of 0.001 at 3000 iterations. Edges are preserved but the filter is failed in regularization, as a side effect of small contrast threshold value. This is because of the amplification of white noise component even it creates small gradient compared to salt & pepper component. Amplification is due to small ratio between contrast threshold and regularization coefficient.

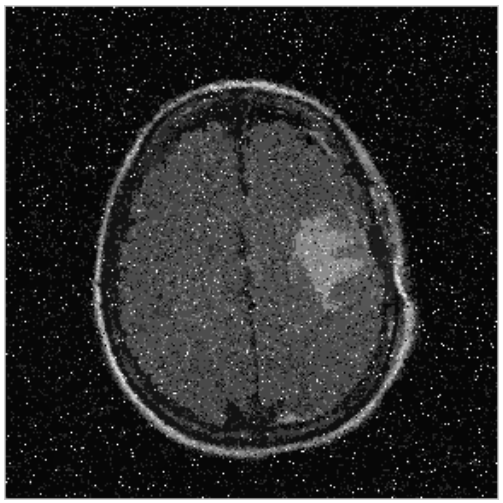

Figure 5.15 – Non-linear isotropic filter output with contrast threshold = 0.001

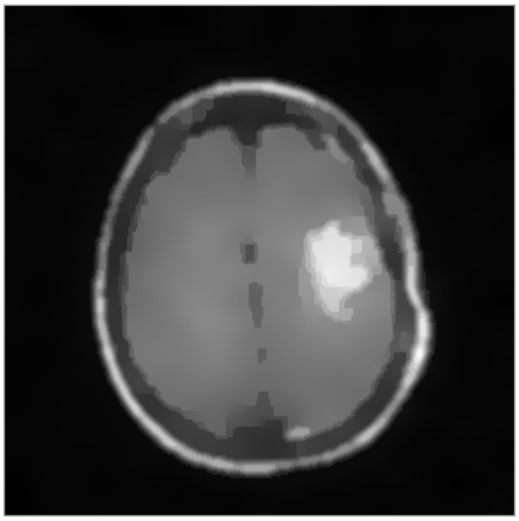

Figure 5.16 – Output with hybrid application of LDF and Perona-Malik

In order to show application of Perona-Malik could be more successful in case of a hybrid use with the linear diffusion filter, another experiment is done. Image is firstly filtered with linear diffusion filter with weighting factor of 0.2 at 10 iterations. Secondly, Perona-Malik filter is applied with regularization coefficient of 0.25 and contrast threshold of 0.0035 for 100 iterations and image given by Figure 5.16 is produced.

5.1.3. Shock Filter

Shock filter is used for sharpening the image, thus, it is appropriate to apply shock filter on blurred images in order to regain the edge information. Mathematically, shock filter is derived by applying heat equation in a time window starting from zero instant and approaching to minus infinity. Solution to that mathematical problem required use of upwind derivatives [**15**].

In order to construct a proper experimental setup, a blurred version of original image on Figure 5.1 is created by using LDF with regularization coefficient of 0.2 at 30 iterations. Created input image is shown on Figure 5.17.

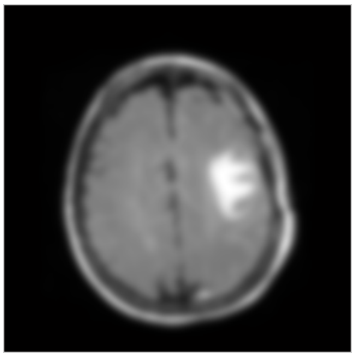

Figure 5.17 – Blurred input for shock filter

Figure 5.18 – Successful output of shock filter

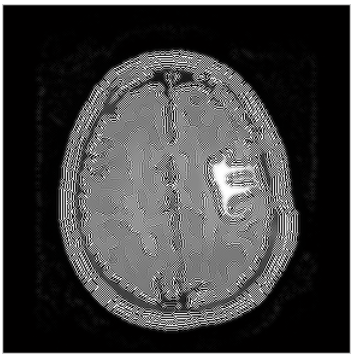

Figure 5.19 – Ringing effect caused by high weighting factor

Shock filter has two inputs similar to LDF case. These inputs are a *weighting constant*, and iteration count. For the first experiment, *weighting constant* is chosen as 0.01 and operation is done with 100 iterations. Output image on Figure 5.18 is produced as a result of this operation. Edges belonging to skull, brain and tumor are reconstructed from blurred image. Right-hand-side figure (Figure 5.19) shows the result of experiment done with a higher *weighting factor* (0.04) at 100 iterations.

As it can be seen on Figure 5.19, main side effect of shock filter is the "*ringing effect*", which occurs as shiny lines following high gradient positions, in case of high weighting factor or high number of iterations.

It has also been observed that, unlike diffusion filters, shock filter is not mean-preserving. Moreover, it is not appropriate to use shock filter for noise elimination because it is prone to amplify the magnitude of gradients. Following figure shows the result of an application of shock filter on the image with salt & pepper noise given on Figure 5.2.

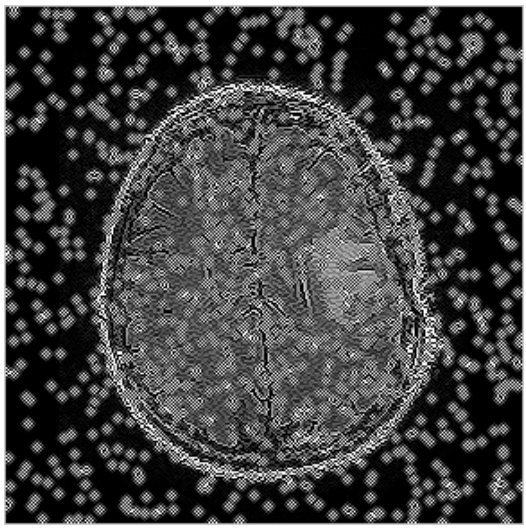

Figure 5.20 – Unsuccessful output of shock filter - noisy image case

5.2. 2D Image Segmentation and Area Computation

2D image segmentation experiments are done with 2 2D brain MRI images. First one is the image given on Figure 0.1, which belongs to a patient with brain tumor. Second image belongs to a patient suffering from multiple sclerosis (MS), symptoms of which is create respectively small, light gray regions over inner region of the brain image. Original image is given by Figure 5.21 and mentioned regions due to MS disease are emphasized by

yellow markers on Figure 5.22. MRI Image is supplied by Prof. Dr. Kader Karlı Oğuz from Dept. of Radiology, in Hacettepe University.

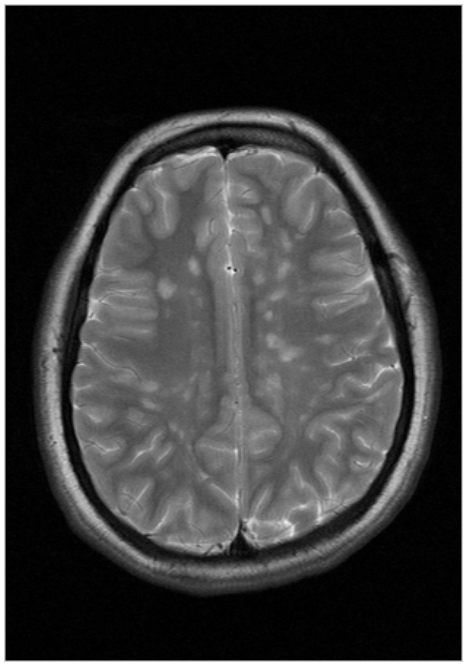

Figure 5.21 –Original brain MRI image with MS (multiple sclerosis)

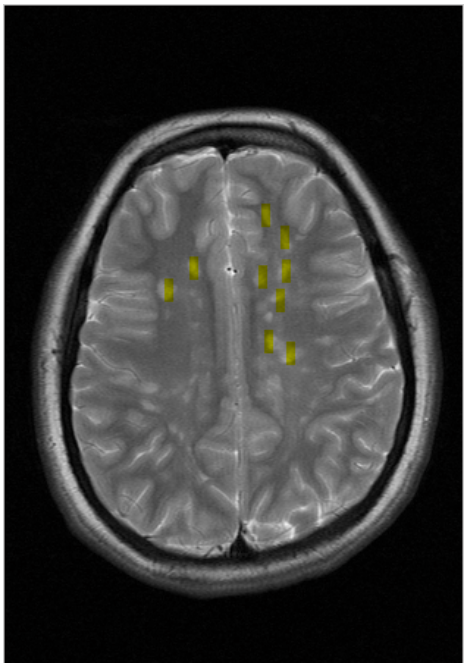

Figure 5.22 – Original image with regions due to MS emphasized

Within the scope of segmentation; input image, selected region of interest, resulting output and corresponding edge map, original image with segment boundaries and binary representation of segmented region are presented by a figure set per experiment. Additionally, numerical inputs and outputs are given in tabular format for each program run.

Results are discussed following the representation of images and numerical information.

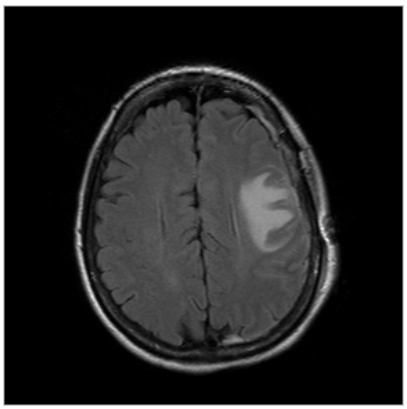

Figure 5.23 – Original brain MRI image with edema [75]

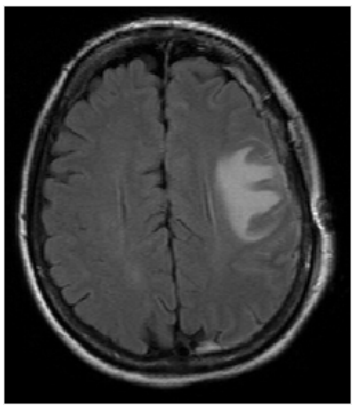

Figure 5.24 – Selected ROI over image domain

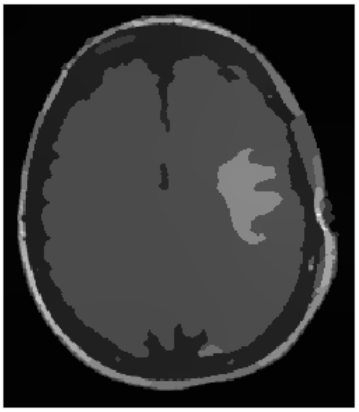

Figure 5.25 – Segmented brain MRI image with edema

Figure 5.26 – Edge map of the segmentation

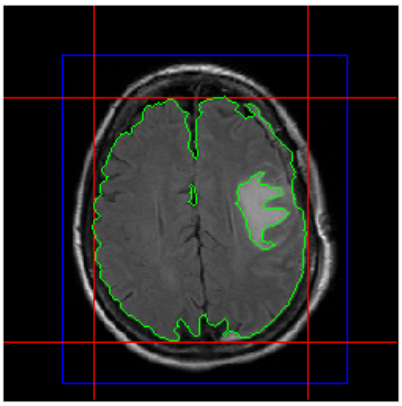

Figure 5.27 – ROI and boundaries

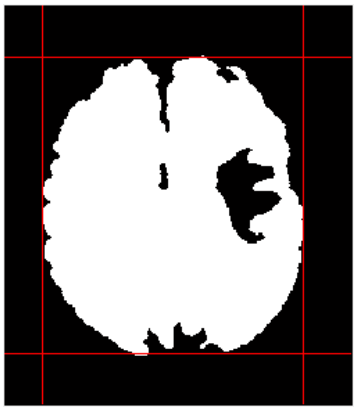

Figure 5.28 – Binary representation of selected area

Input image (Figure 5.23) is narrowed down with selection of a proper ROI. For this program run, ROI is selected to cover all of the meaningful

information in the image; however, partial elimination of zero intensity background provides computational speed for the application.

As shown by Figure 5.25, application manages to partition the image into semantically meaningful smooth sub-regions. Boundaries of brain and tumor can easily be examined from the edge map given by Figure 5.26 and they are marked by green curve on Figure 5.27. Final figure of above figure group shows the image acquired by applying region growing with a seed point included inside the boundary surrounding the brain, but outside the tumor region.

In order to be able to compute area by using 2D image, converting the image into binary form and dividing the domain into two partitions – selected region and complementary region – correctly is crucial. Therefore, it is important at first step of segmentation to achieve "*cartoonization*" with minimum variations inside partitions of the image. In this condition, locations of the edges does not depend on user-defined region growing parameters such as "*neighborhood radius*" and "*threshold range*", and region growing is used just for extraction of already defined region from the image.

Table 5.1 shows the numerical values for inputs and outputs of this experiment. All three inputs given are Ambrosio-Tortorelli parameters, which can be defined as the weighting factors for three terms of the energy functional to be minimized. For this particular program run, inputs are selected intentionally to have successful results. Effects of changing *regularization coefficient* and *edge complexity factor* coefficients are investigated in following subsection of this section.

As it can be seen from both Table 5.1 and Figure 5.29, mean value of the intensities of pixels in selected ROI does not change with iterations, as expected. Segmentation operation produced decay in both standard deviation and entropy of the selected ROI over the image, which is because of elimination of high frequency variations over partitions, which are composed of *noise* and *texture*.

Table 5.1 – Numerical information regarding segmentation of image with tumor

INPUTS	**REGULARIZATION COEFFICIENT**	100	
	DATA FIDELITY FACTOR	10	
	EDGE COMPLEXITY TERM	0.05	
OUTPUTS	**ITERATION COUNT**	20000	
	RATE OF CHANGE OF SSD	$1.2365e^{-003}$	$9.6946e^{-006}$
	MEAN VALUE (ROI)	0.1952	0.1952
	STANDARD DEVIATION (ROI)	0.1642	0.1563
	ENTROPY (ROI)	6.09	4.189
	TOTAL ENERGY	$2.263e^{005}$	$6.231e^{004}$

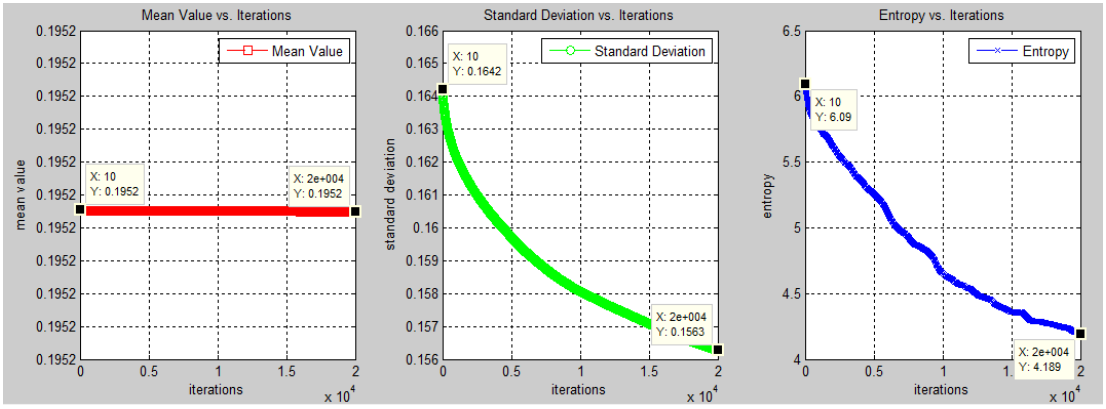

Figure 5.29 – Statistical evolution regarding segmentation of image with tumor

Since entropy can be defined as a measure of information carried by the image, the segmentation operation seems to reduce the amount of total information with increasing number of iterations.

Although at first glance “reducing the amount of information” sounds odd, this is just what is necessary to be able to extract useful information from the domain. Therefore, in image segmentation sense, it would not be improper to say that “less information is more information”.

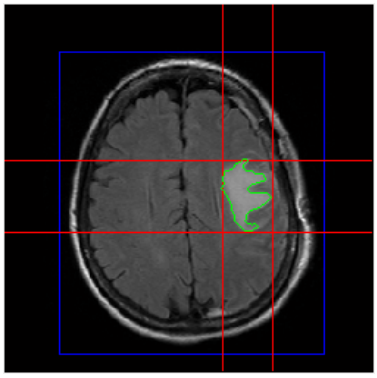

Figure 5.30 – ROI and boundaries for edema region

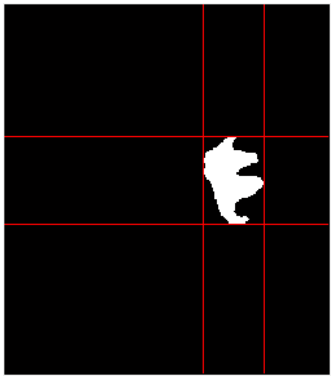

Figure 5.31 – Binary representation of edema region

Two images given above are produced by using same set of input parameters, but just changing the seed point of region growing applied at the last step. This time, region representing tumor is selected and extracted from the input.

Table 5.2 – Numerical information regarding distance and area measurement

METADATA	**PIXEL X-SIZE**	0.79861 mm
	PIXEL Y-SIZE	0.79861 mm
	SLICE THICKNESS	5 mm
COMPUTATIONS	**MAXIMUM DISTANCE ALONG X-AXIS**	40 px
		31.9444 mm
	MAXIMUM DISTANCE ALONG Y-AXIS	57 px
		45.5208 mm
	CROSS-SECTIONAL AREA	1154 px
		735.9978 mm^2
	VOLUME ON SLICE	3679.9888 mm^3

Using the relevant fields of metadata read from the DICOM image file, sizes of the projections of tumor over horizontal and vertical axes, cross-sectional area of the tumor, and the volume belonging to tumor over that particular MRI slice are computed and results are presented in Table 5.2.

Second experiment of this section is done on the 2D brain MRI image belonging to a subject suffering from MS disease, as mentioned before. This case can be said to be more challenging compared to the tumor case, since the target object(s) has (have) respectively lower gradient magnitude at boundaries, and has (have) smaller area than the tumor cross-section on the first experiment has.

In order for the iterations to be terminated before meaningful information (regions belong MS lesions) is lost by regularization, lower limit for the rate of change of sum of squared difference is raised to $1.0e^{-005}$ from $1.0e^{-006}$. Therefore, the operation lasted for less than 20000 iterations.

Figure 5.37 shows binary representation of selected area. Black regions surrounded by the white region represent the segmented MS lesions. It can be observed from Figure 5.37, meaningful results which can be used for diagnosis and prognosis purposes, such as total number of lesions, total area of lesions, area of a particular lesion, can be computed by arranging the appropriate seed point location.

Key difference of the implemented segmentation algorithm from widely-used techniques such as "*region growing*" and "*thresholding*" is the effects of changing the inputs over the resulting output. In those traditional methods, size and edge positions of distinct regions directly depend on the manual preferences; however, varying "*Mumford-Shah parameters*" is not observed to change the edge positions; instead regularization rate inside distinct regions, number of resulting distinct partitions, or amount or complexity of edge information is affected by applying the algorithm with different input combinations.

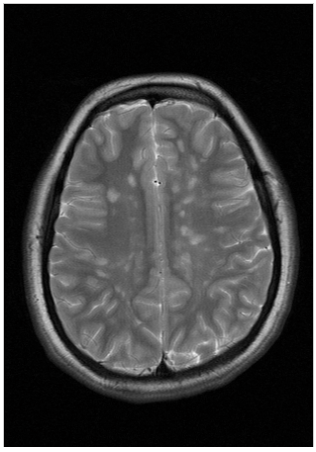

Figure 5.32 – Original brain MRI image with multiple sclerosis (MS)

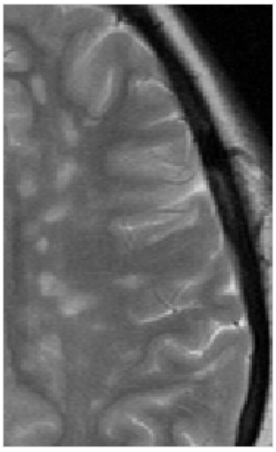

Figure 5.33 – Selected ROI over image domain

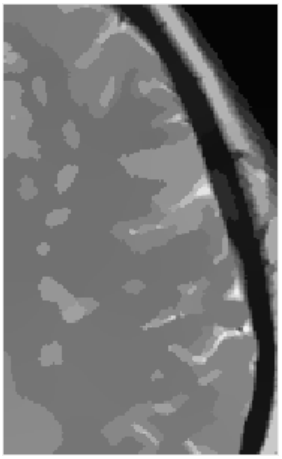

Figure 5.34 – Segmented brain MRI image with MS

Figure 5.35 – Edge map of the segmentation

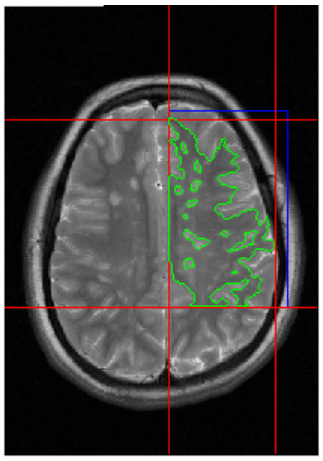

Figure 5.36 – ROI and boundaries

Figure 5.37 – Binary representation of selected area

Region growing technique is only employed to extract the target region from the already segmented image with a small threshold range; therefore, does not put a constraint over the boundary locations. This is because high gradient is already accumulated at boundaries of the image, and high frequency variations over boundaries are eliminated by the regularization property of Mumford-Shah segmentation.

Numerical inputs and outputs of the operation are given in Table 5.3. Same set of segmentation parameters are used for this experiment. Iterations are terminated at the first time when rate of change of SSD drops below $1.0e^{-005}$, without waiting for the iteration count to reach the maximum number of iterations. Similar results are acquired for the statistical outputs of the process.

Table 5.3 – Numerical information regarding segmentation of image with MS lesions

INPUTS	**REGULARIZATION COEFFICIENT**	100	
	DATA FIDELITY FACTOR	10	
	EDGE COMPLEXITY TERM	0.05	
OUTPUTS	**ITERATION COUNT**	8641	
	RATE OF CHANGE OF SSD	$8.855e^{-004}$	$9.4959e^{-006}$
	MEAN VALUE (ROI)	0.4296	0.4296
	STANDARD DEVIATION (ROI)	0.1745	0.1589
	ENTROPY (ROI)	6.9666	5.5056
	TOTAL ENERGY	$1.703e^{005}$	$3.746e^{004}$

5.2.1. Analysis of Relation between Ambrosio-Tortorelli Energy and Input Parameters

In this subsection, effects of input parameters of Ambrosio-Tortorelli segmentation method are analyzed. Analysis is done by the observation of values of the data fidelity term of minimized cost functional (Equation **(A.1)**) for varying α, β, and, ρ values. For each input combination, segmentation operation is done in 1000 iterations. Result is introduced as a surface plot of SSD values depending to the input combinations. On the surface plot, x-axis shows values for the *regularization coefficient*, and y-axis shows values for the *edge complexity factor*. *Data fidelity coefficient* is kept constant at a value of 1.

First metric is designed as "*sum of squared differences (SSD)*" between segmentation output and a previously segmented reference image. Inputs producing small values of SSD are expected to give the best match to the reference image.

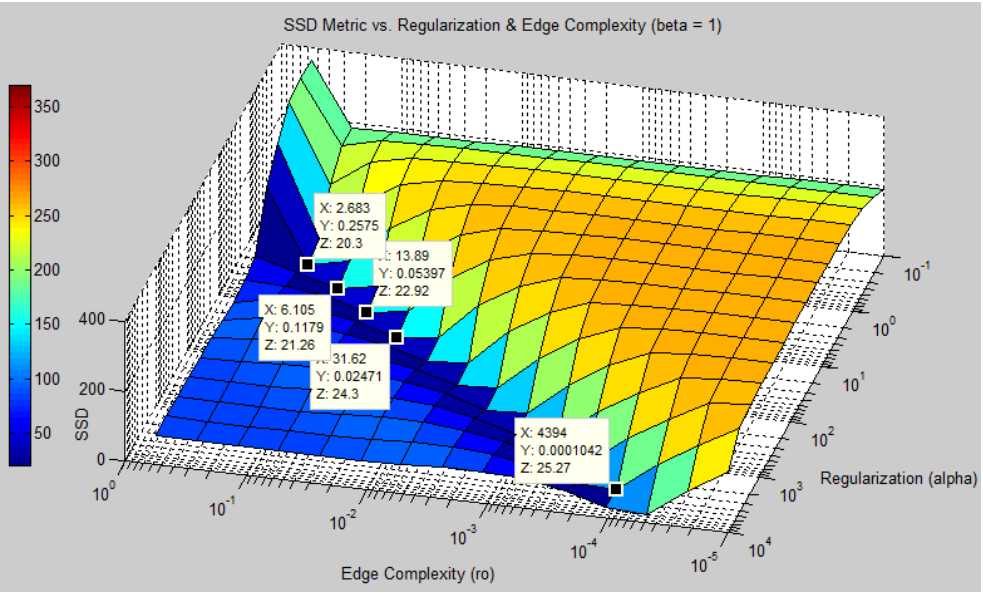

Figure 5.38 – Data Fidelity Metric vs. Regularization & Edge Complexity (beta=1)

Figure 5.38 shows values of data fidelity metric at grid positions belonging to input combinations. 5 minimum points are marked on the figure. Corresponding regularization coefficients, edge complexity terms, multiplications of inputs, and SSD values are presented in Table 5.4. It is observed that $\alpha \times \rho$ falls into a narrow domain (0.4394 - 0.7810) for inputs producing minimum SSD values.

Table 5.4 – 5 minimum points of SSD plot

(beta = 1)	alpha	ro	alpha x ro	SSD
1st	2.683	0.2575	0.6909	20.3
2nd	6.105	0.1179	0.7198	21.26
3rd	13.89	0.0540	0.7501	22.92
4th	31.62	0.0247	0.7810	24.3
5th	4394	0.0001	0.4394	25.27

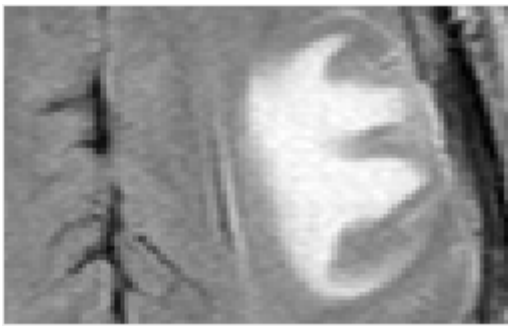

Figure 5.39 – Input image

Figure 5.40 – Reference segmentation

Figure 5.41 – Output corresponding to the 1st minimum

Figure 5.42 – Output corresponding to the 5th minimum

Figures from Figure 5.39 to Figure 5.42 show the input (non-segmented) image, previously segmented reference image, output (segmented) image corresponding to the input set producing 1st minima for SSD, and, output (segmented) image corresponding to the input set producing 5th minima for SSD. Output images produced by the input combinations over minimum path of the convex surface of SSD values are empirically close to the reference segmented version of image.

5.3. 3D Image Segmentation and Volume Computation

A separate module has been implemented for the purpose of viewing volumetric images, and applying 3D segmentation algorithm on corresponding images.

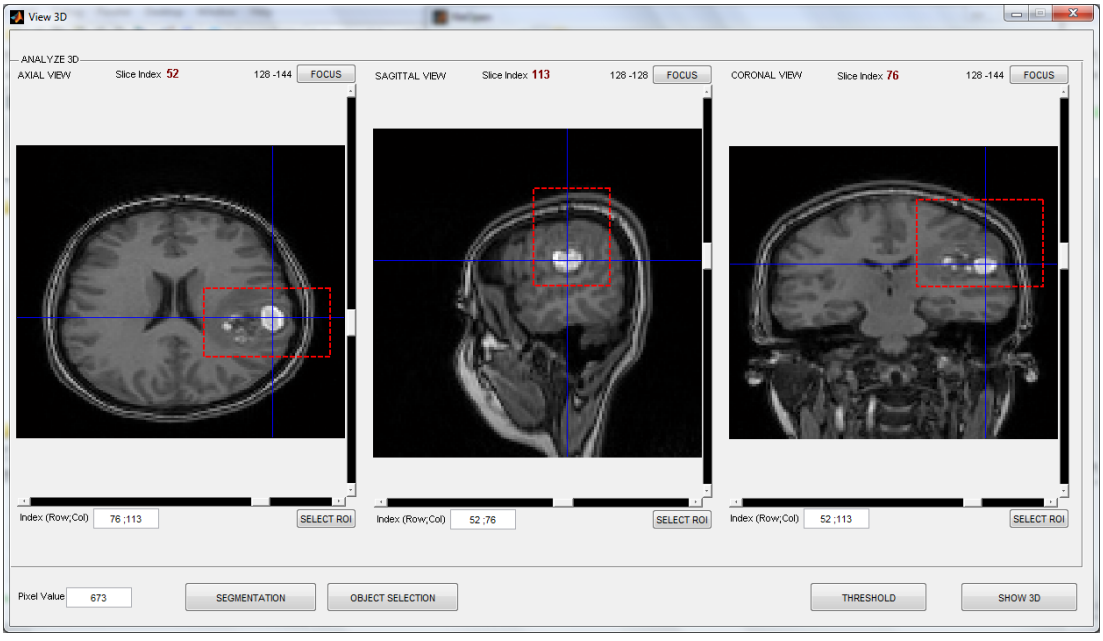

Figure 5.43 – A screenshot image of 3D Medical Image Viewer module

A screenshot of the main window belonging to implemented 3D Medial Image Viewer module is given by Figure 5.43. Volumetric brain MRI data is supplied by Prof. Dr. Ayşenur Cila from Dept. of Radiology in Hacettepe University. Axial and Coronal projections are reproduced using the actual data - sagittal slices shown in the centre. Focused point and rectangular volumetric ROI in 3D space can be selected over any one of the projections. Coordinates of the focused point is represented by blue lines, and boundaries of the selected volumetric ROI are represented by dashed red lines.

Volumetric segmentation is applied on selected ROI and intensity values for the voxels falling inside the ROI is replaced with the segmentation output, after the segmentation operation is completed.

5.3.1. Volumetric Segmentation

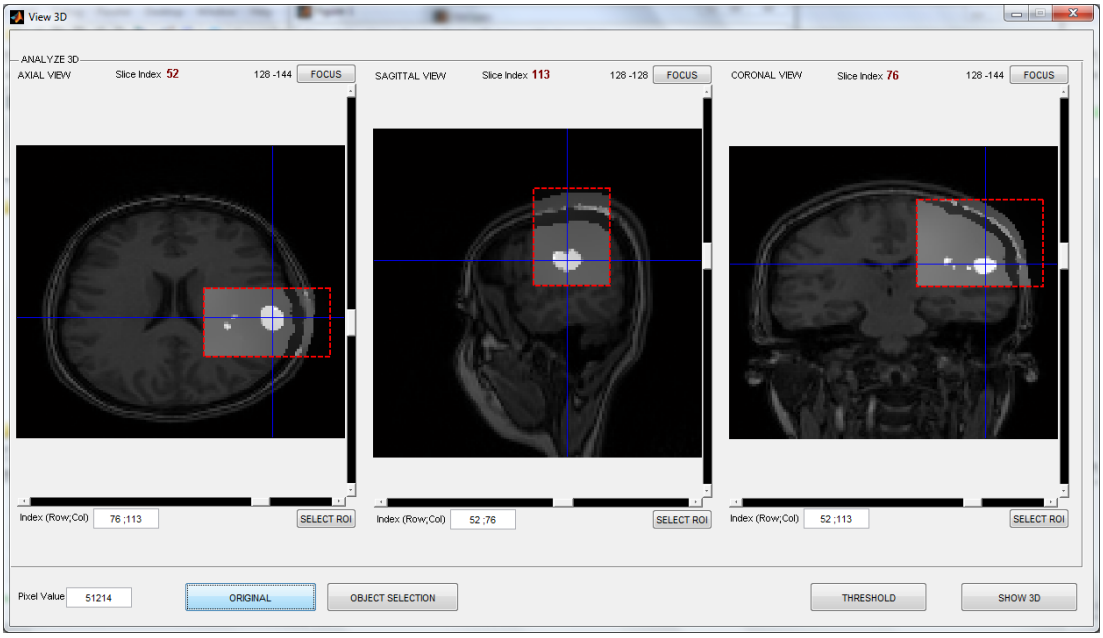

Figure 5.44 – 3D view of the segmented volumetric image

3D segmentation is applied on the volumetric image shown by Figure 5.43. Figure 5.44 shows the screenshot of 3D Viewer main window after volumetric segmentation. As it can be perceptually observed from all projections, tumor, brain and skull seem to be separated into distinct regions. Ambrosio-Tortorelli energy is reduced from $3.074e^{005}$ to $1.208e^{005}$ (reduced by 61%).

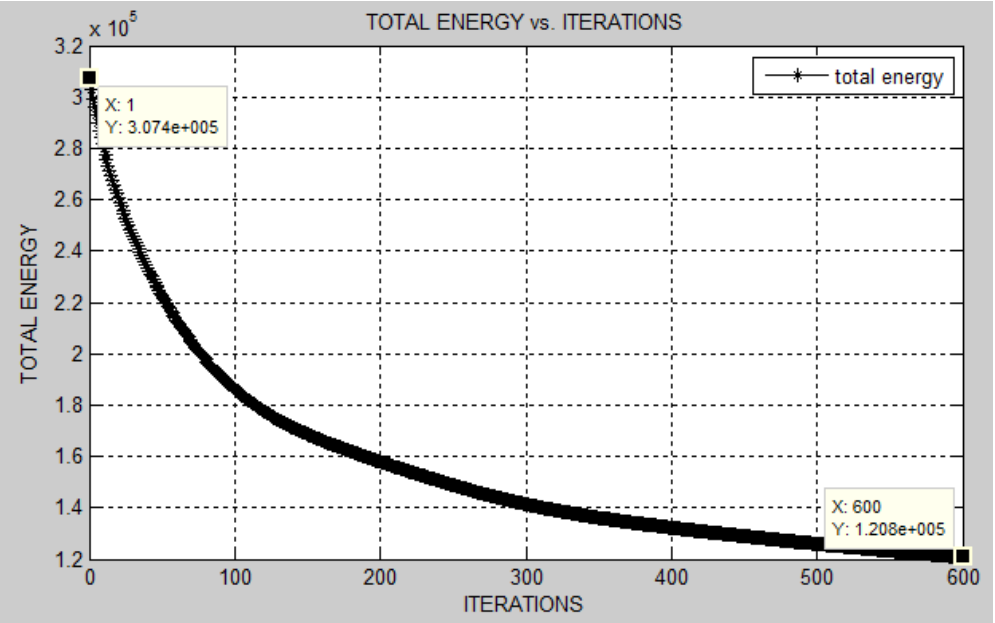

Figure 5.45 – Graphical representation of energy vs. iterations

A volumetric visualization of the object including the focused point is done by using 3D region growing (it can also be called as *volume growing*) algorithm. Single object is selected by 3D region growing; it is visualized together with its voxel size, related metadata information, and computed volume. Figure 5.45 shows the evolution of the values of the cost function due to iterations. Figure 5.46 shows the volume representation of selected object. Table 5.5 gives the numerical data regarding the volume computation.

Table 5.5 – Data regarding volume computation

Sagittal voxel size	1.875 mm	
Coronal voxel size	1.875 mm	
Axial voxel size	1.3 mm	
Volume	409 voxels	1437.8906 mm^3

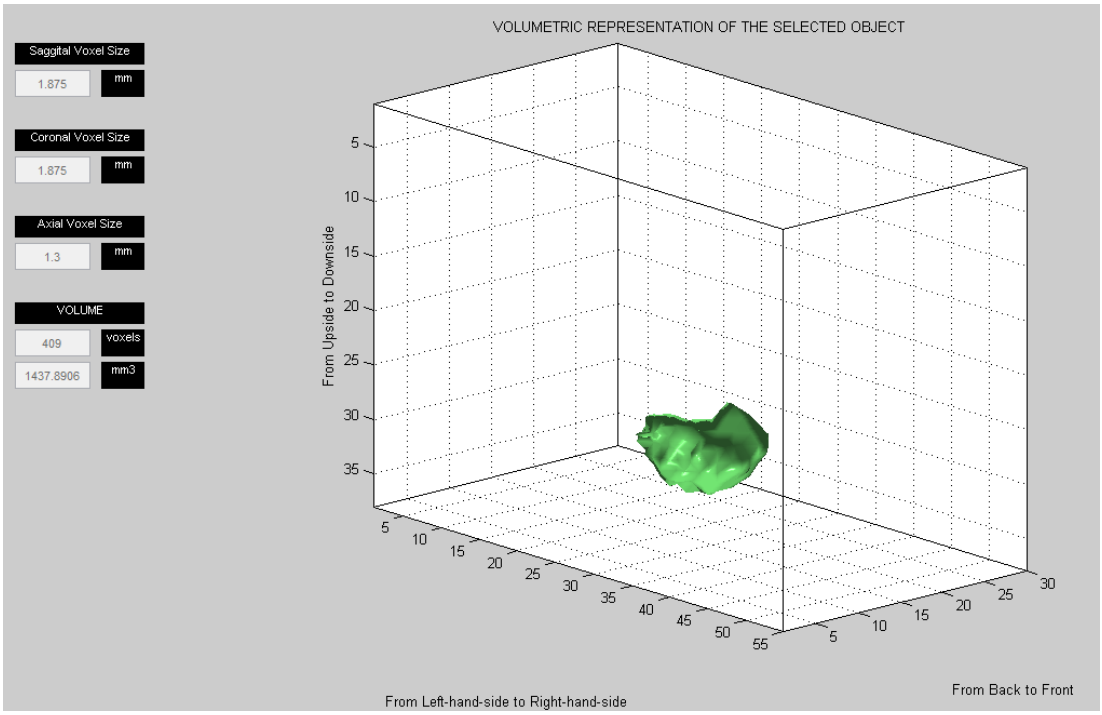

Figure 5.46 – Volumetric representation of the selected object (tumor).

5.3.2. Volume Visualization

In addition to the volume computation purpose, module can be used for viewing segmented or non-segmented data, with thresholding over 3D model view.

Two features are employed for this purpose. First one of these is the surface painter, which is seen on above figure in green color, and the second one is the edge drawer, which is pink and is in form of a mesh grid.

Threshold values for both edges and the object itself can be manually arranged over the model view. Examples of volume visualization within two different ROI are presented by following two figures.

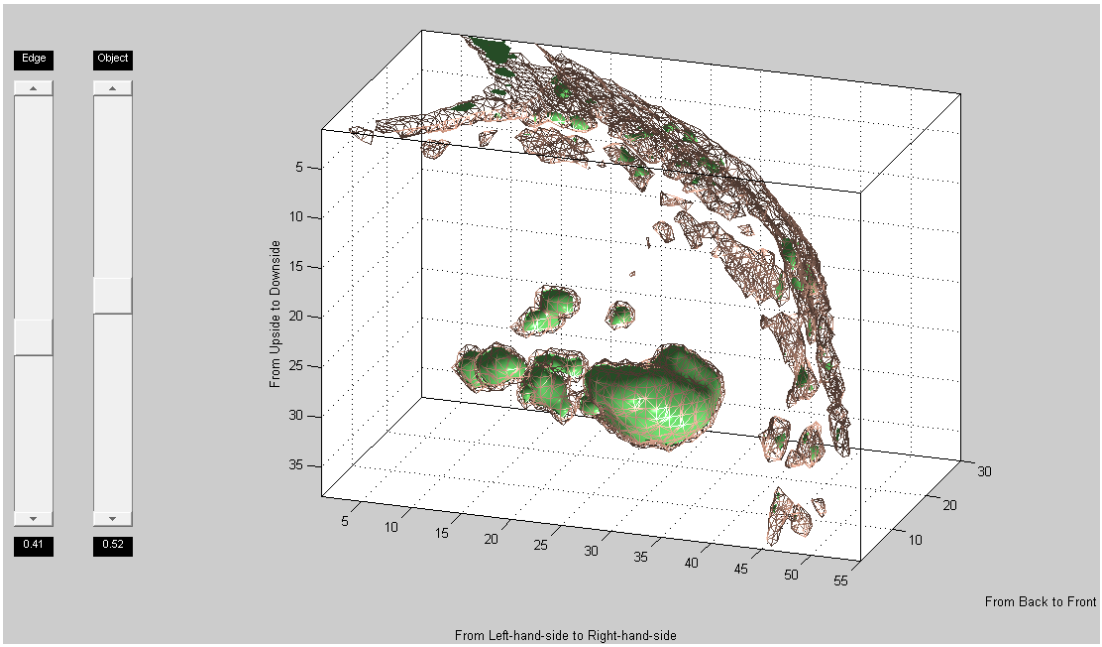

Figure 5.47 – Volume visualization example-1

Outer surface of skull is shown as "*edge*" by pink grid lines, and outer surface of the tumor regions are shown as "*object*" with green color in both of the examples.

In second example a larger ROI is selected, and the azimuth angle of view is changed.

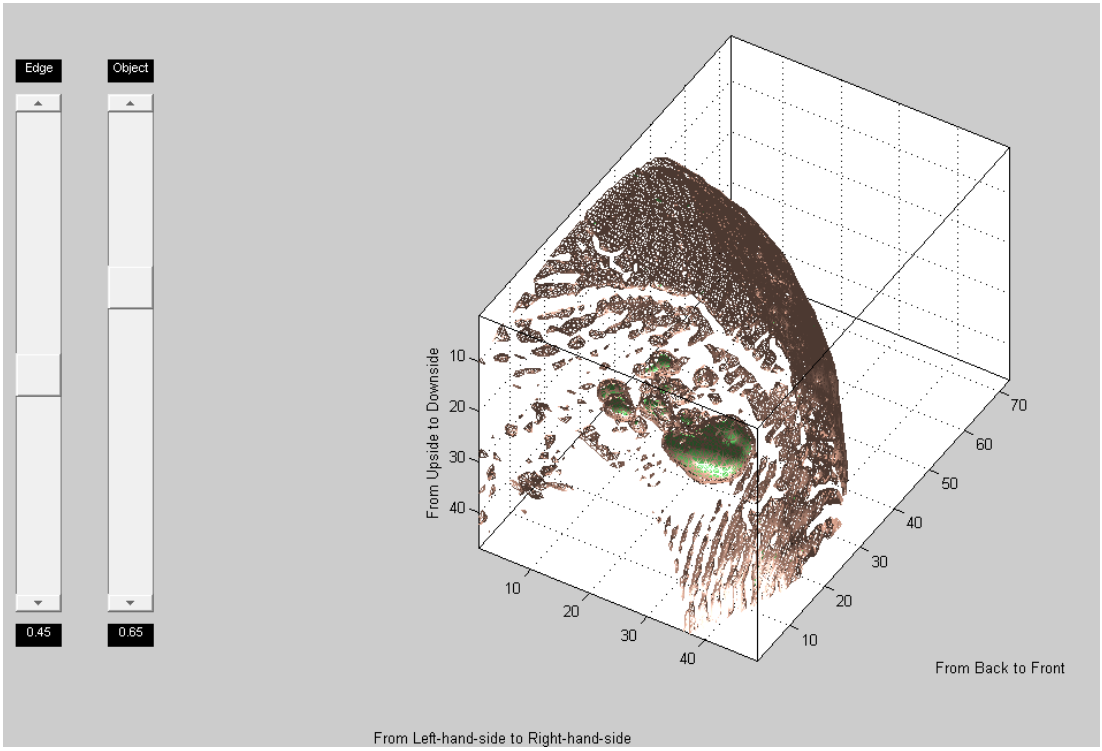

Figure 5.48 – Volume visualization example-2

5.4. 2D Image Registration

5.4.1. Fully-Automated 2D Rigid Registration with Scale Parameters

Fully-automated registration of brain MRI images requires minimization of minus of normalized cross correlation between the reference and the corrected image, by finding the transformation with appropriate input parameters. For 2D images, there is 1 input parameter for rotation, and there are 2 input parameters for each one of translation and scaling to be searched for.

Demonstration of fully-automated 2D rigid registration is prepared by the application of the algorithm on segmented versions of brain MRI images, acquired at different time instants, from the same subject. On one of the images, there exists a region showing the cross section of edema occurred because of a tumor. In the other image, that region does not appear. Segmented images are given by Figures 5.49 and 5.50.

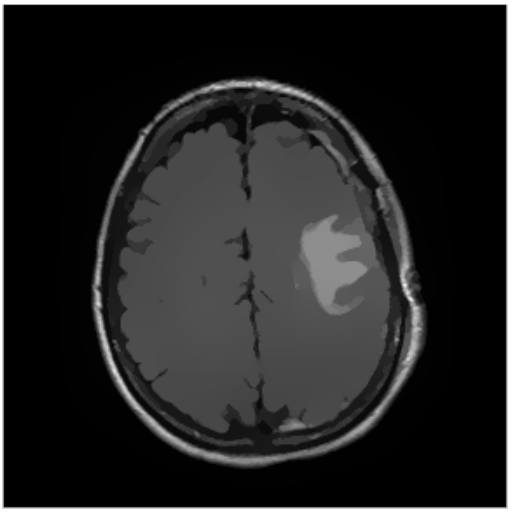

Figure 5.49 – Segmented brain MRI image with edema

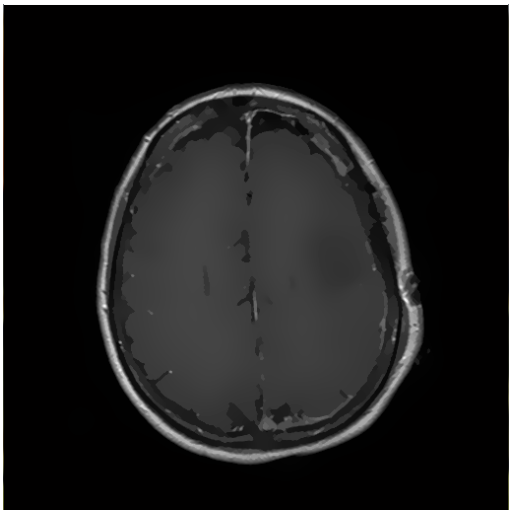

Figure 5.50 – Segmented brain MRI without edema

Table 5.6 – Initial and final values for input parameters

Parameters	**Rotation (degrees)**	**X Translation (pixels)**	**Y Translation (pixels)**	**X Scaling**	**Y Scaling**
Initial Value	1.0	1.0	1.0	1.0	1.0
Final Value	-0.2191	6.698	12.04	0.9704	0.9628

Image given by the right-hand-side figure is selected as the reference image, and the other image (given by left-hand-side figure) is selected as the input image to be corrected. Reference image has 512 rows and 512 columns, where the moving image has 288 rows and 288 columns. Image intensities are normalized in the range of 0.0 and 1.0. For the 2D registration

experiment, initial values of the input parameters are assigned arbitrarily as given in Table 5.6.

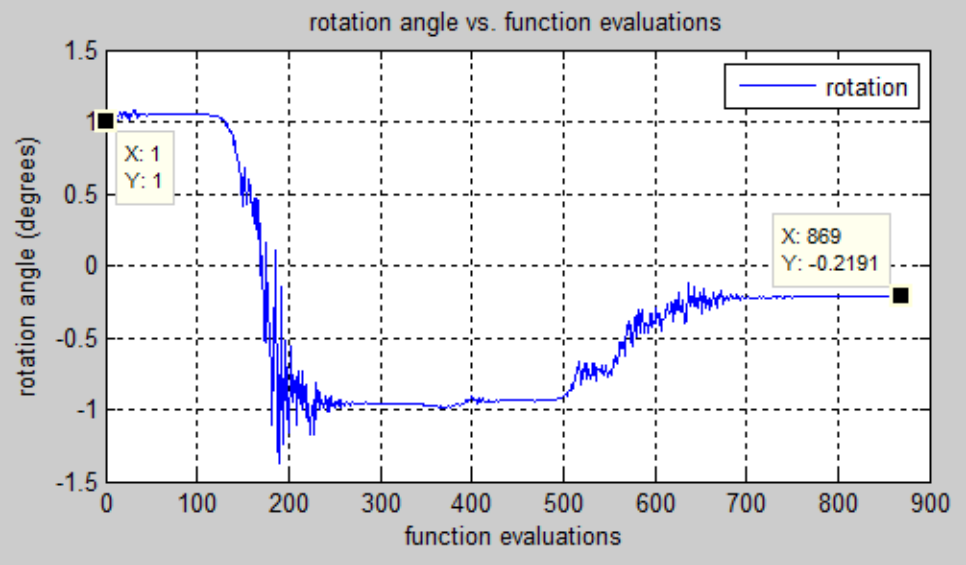

Figure 5.51 – Trajectory of rotation angle

Trajectories followed by the parameters during the function evaluations are determined by the Nelder-Mead Simplex algorithm implementation in MATLAB® Optimization Toolbox, which is named as "*fminsearch*". Trajectories of rotation angle, translation parameters, and scaling parameters are presented by Figures from Figure 5.51 to Figure 5.53. Also, data tips showing initial and final values of the variables are inserted into graphical figures for each parameter.

Normalized cross correlation takes values between 0 and 1. Since employed algorithm aims to minimize the determined metric, additive inverse of normalized cross correlation value is used as the measure of similarity. That means, ideal value for our metric is -1, which would be the case between two identical images.

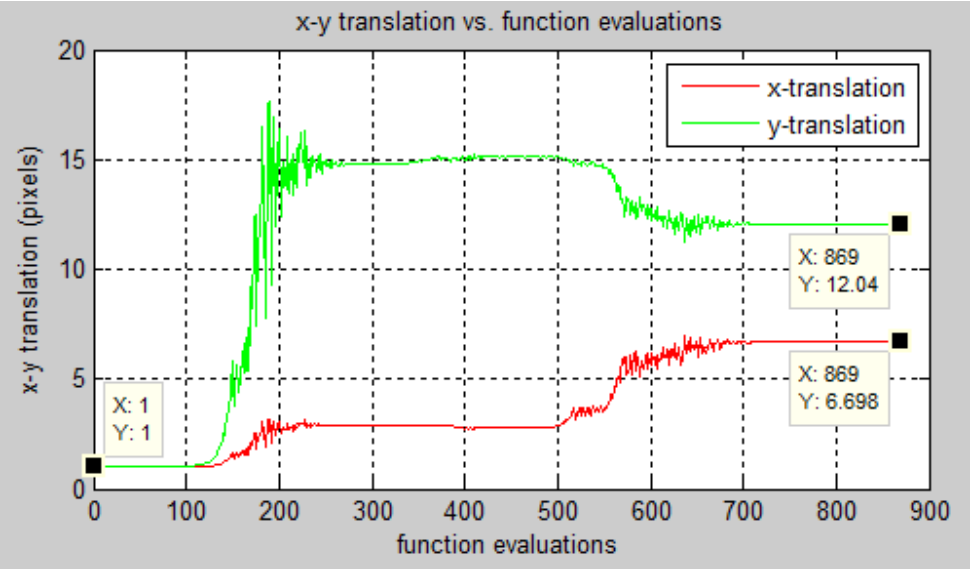

Figure 5.52 – Trajectory of translation parameters

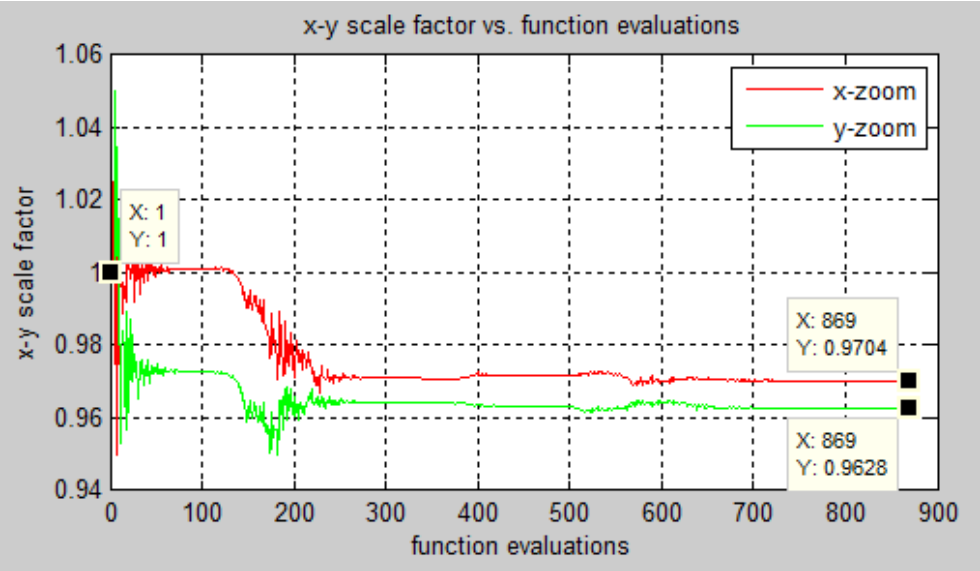

Figure 5.53 – Trajectory of scaling parameters

Application of the transformation with initial parameters on the moving image creates a value of -0.7348 for the similarity metric. As shown by Figure 5.54, value of the metric is reduced at each update of transformation parameters. Finally, when rate of change of the metric value falls below the error

tolerance, -0.9346 is accepted as the final value of parameter search. Moving image is interpolated to make the two images agree in dimensions, and transformation model with found parameters is applied on it.

Used algorithm is not designed in order to find the global minimum value; therefore, there always exists a probability of getting stuck in local minimums. This situation is dependent on the initial values of input parameters.

Absolute value of difference of reference and the moving image is visualized before any change is applied over the input image and after the transformation is applied with appropriate inputs. Resulting images are given by Figure 5.55 and Figure 5.56. It is obviously seen from Figure 5.55 that without registration, neither skull nor the boundaries of brain of the subject is aligned in different images. Non-existence of those regions and existence of the region representing the edema in the second image gives an idea of the success in 2D registration operation.

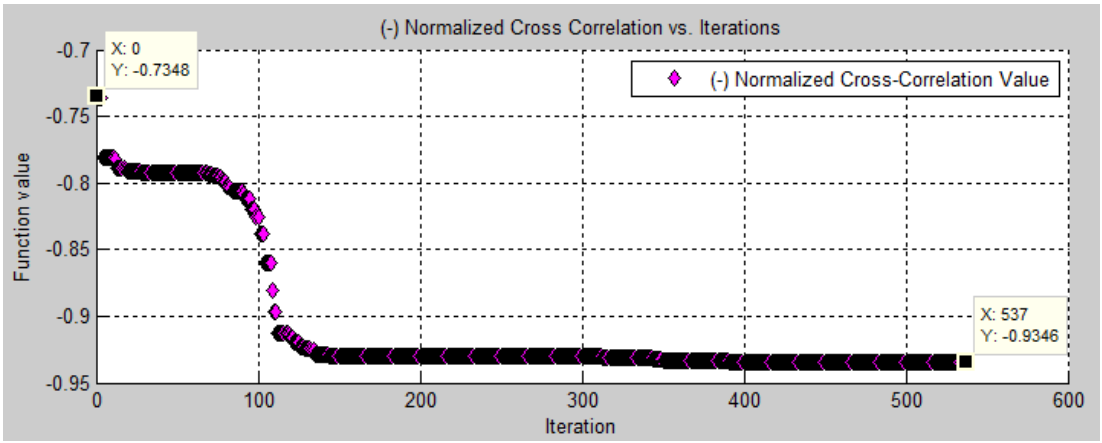

Figure 5.54 – Evolution of minus normalized cross correlation

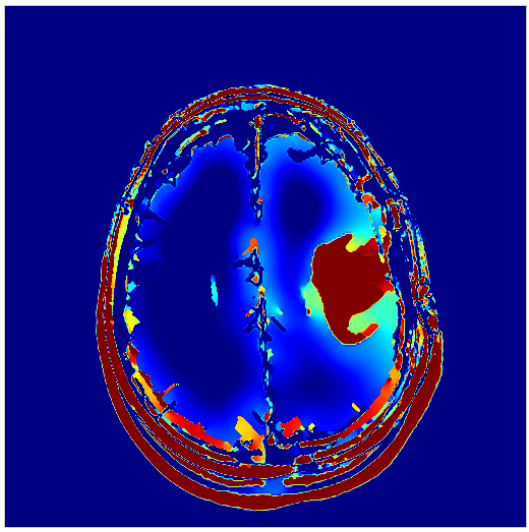

Figure 5.55 – Absolute difference between reference image and the initial version of moving image (input image)

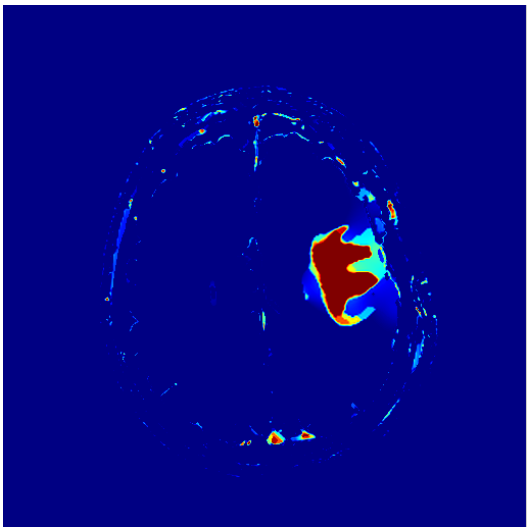

Figure 5.56 – Absolute difference between reference image and the final version of moving image (corrected image)

5.5. 3D Image Registration

5.5.1. Fully-Automated 3D Rigid Registration with Scale Parameters

3D registration experiment is done with aligning 2 different 3D head models, reconstructed from volumetric coronal MRI datasets belonging to the same subject, acquired at different time instants. Reference head model, input data to be aligned to the reference, and absolute difference of these two before registration is visualized and presented by figures Figure 5.61, Figure 5.62, and Figure 5.64, respectively. Two sets of volumetric MRI images are provided by Prof. Dr. Ayşenur Cila from Dept. of Radiology in Hacettepe University. As it can be seen on third figure, absolute difference, thus, squared sum of differences between reference image and the initial state of moving image (minimization of which can be used as a metric in image registration) is respectively high.

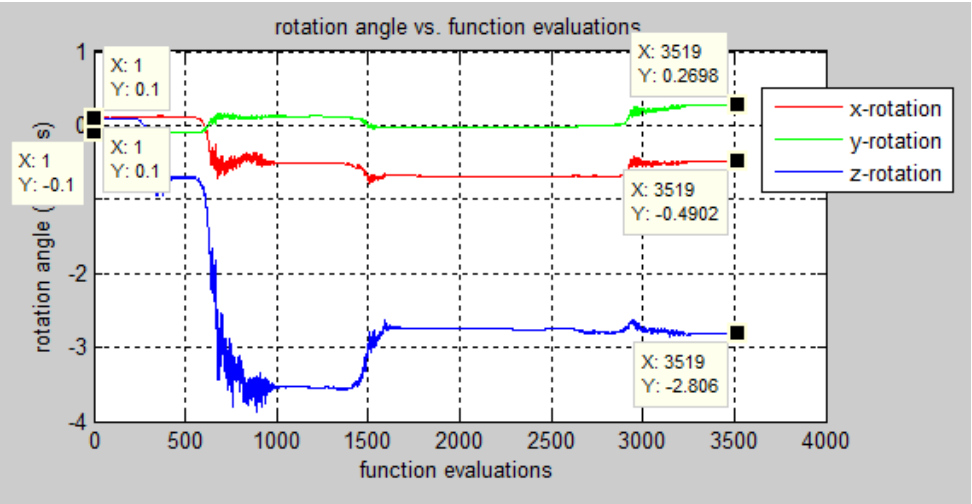

Figure 5.57 – Trajectory of rotation parameters

Results are presented in a similar form with the ones in the previous section, fully-automated 2D registration case. Figures from Figure 5.57 to Figure 5.59 show the trajectories followed by the input parameters of transformation matrix. Inputs are grouped into three by their modification properties: rotation (3 parameters), translation (3 parameters), and scaling (3 parameters).

Inputs are selected arbitrarily, considering smallness in deviation from assumption of two identical images. Starting values of rotation and translation parameters are assigned as 0 ± 0.1 and starting values of scaling parameters are assigned as 1 ± 0.1.

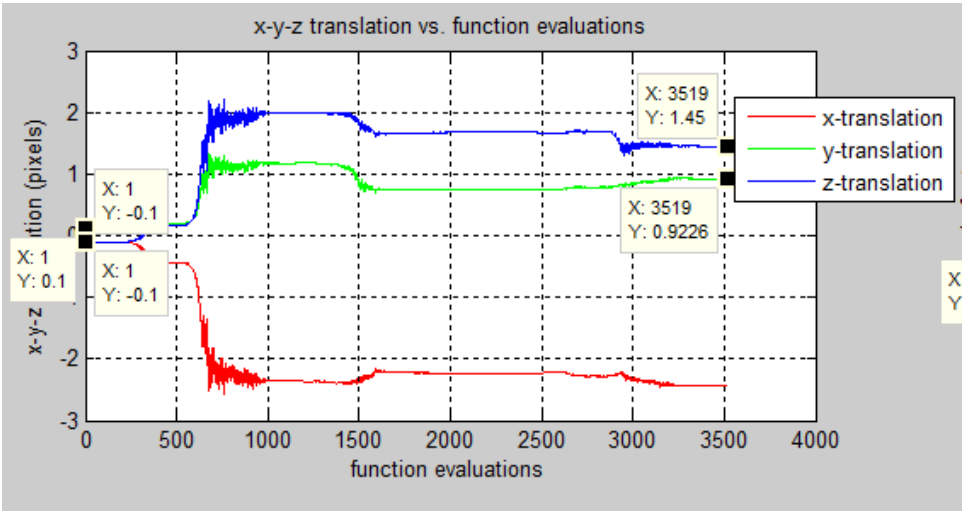

Figure 5.58 – Trajectory of translation parameters

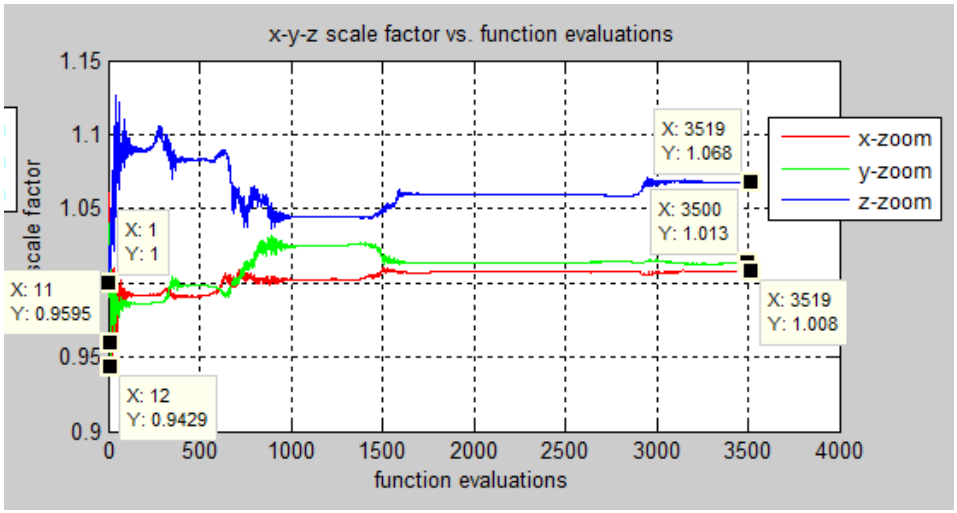

Figure 5.59 – Trajectory of scaling parameters

Shape of parameter search trajectories visualizes the "*reflection*", "*expansion*", "*contraction*" and "*shrinking*" operations of *Nelder-Mead Simplex Method*.

Figure 5.60 shows the evolution of similarity metric - additive inverse of 3D normalized cross correlation - between reference and moving images. Normalized cross correlation is increased from 0.7714 to 0.8464, by 7.5%.

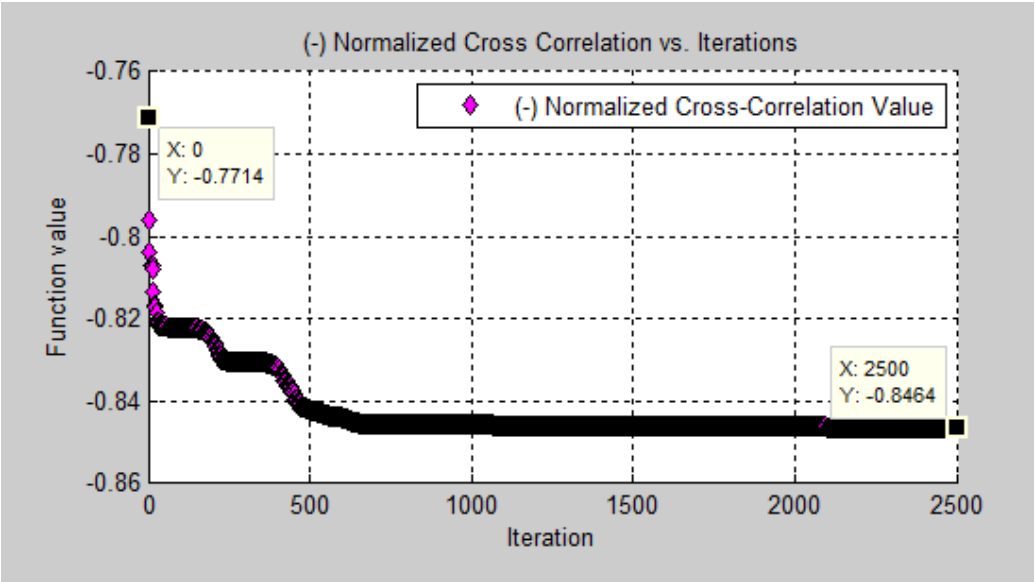

Figure 5.60 – Evolution of 3D minus normalized cross correlation

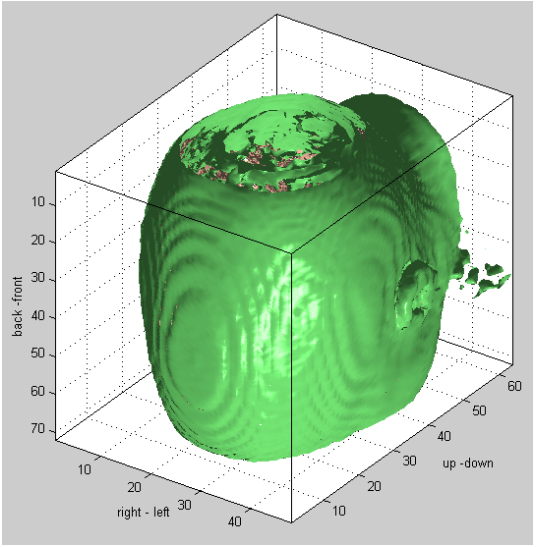

Figure 5.61 – Reference 3D image

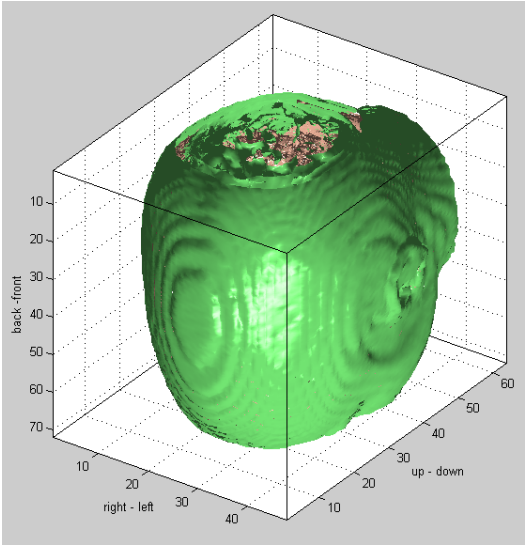

Figure 5.62 – Initial 3D input image

Figure 5.63 shows the corrected version of the moving image which is the output of 3D registration operation. Although it is not so easy to get a perceptual idea of the amount of success by looking at this image, it is possible to have the idea with the image given by Figure 5.65, which shows

the absolute difference between reference and output images. Comparing this with Figure 5.64 shows the operation is working towards desired direction.

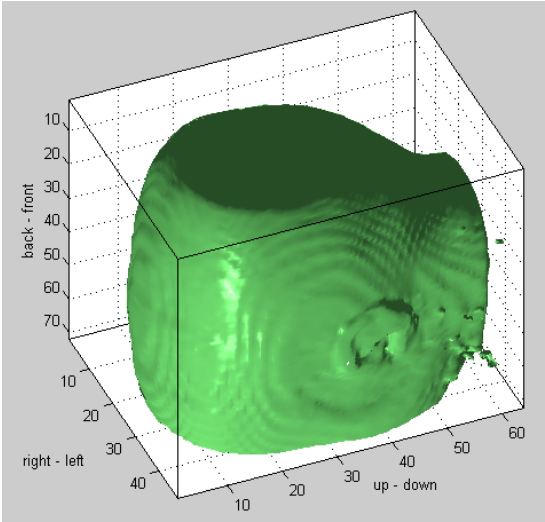

Figure 5.63 – Corrected (registered) version of the moving image

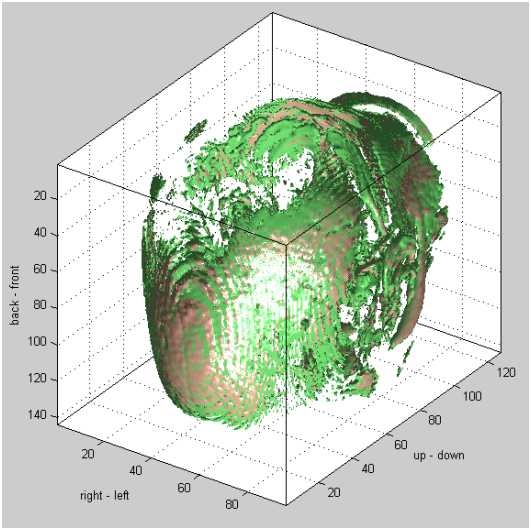

Figure 5.64 – Visualization of absolute difference between reference and input images

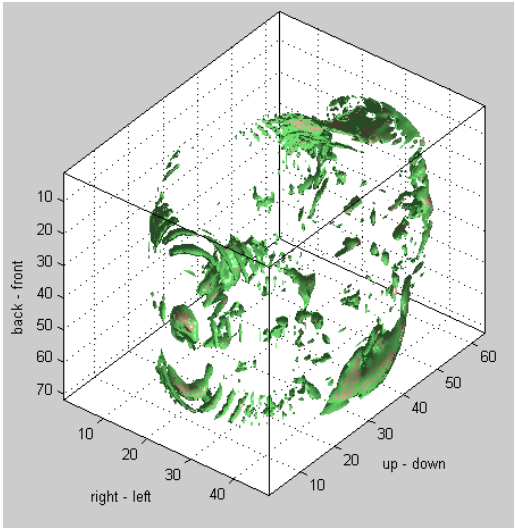

Figure 5.65 – Visualization of absolute difference between reference and corrected images

Starting and final values of the transformation parameters and the similarity metric is presented on following table. It is observed that sign of the parameters does not remain constant during search process, which is consistent with the theory of Nelder-Mead Simplex Method.

Table 5.7 – Initial and final values for input parameters

Parameters	Initial Values	Final Values
Rotation around x-axis	0.1 degree	-0.49021 degree
Rotation around y-axis	-0.1 degree	-2.8063 degree
Rotation around z-axis	0.1 degree	0.26979 degree
Translation along x-axis	-1 pixels	-2.4236 pixels
Translation along y-axis	1 pixels	0.92256 pixels
Translation along z-axis	-1 pixels	1.4499 pixels
Scaling along x-axis	1.01	1.0079
Scaling along y-axis	0.99	1.0129
Scaling along z-axis	1	1.0679
METRIC	-0.7714	-0.8464

CHAPTER 6

CONCLUSION

6.1. Conclusions

Medical imaging is one of the key fields of biomedical engineering, which aims to apply engineering principles in field of medicine and biology. Cumulative development of medical imaging science provides intense improvement in diagnosis, prognosis, and therapy. Accuracy in analysis of medical imaging data is at least as important as reliability of data acquisition process. Analysis of medical imaging data requires application of techniques involving image processing, which is one of the most studious topics of engineering and computer science.

Main purpose of this thesis study is to perform a comprehensive review of image processing literature, and to implement a generic application framework by using the infrastructure retrieved from the literature review. It is aimed for the application framework to be composed of independent modules which enable half or full automation of common routines and procedures used in analysis of medical images by radiologists. In consideration of the

literature review and field analysis, main modules and hierarchical structure behind those have been designed.

Literature review is performed involving three main groups of image processing, which also constitute main modules of implemented software. These three groups can be listed as image filtering, image segmentation and image registration. As well as mathematical principles behind these main concepts are explored, relations among them are investigated throughout the research progress. Consequently, corresponding modules are implemented to be controlled by a main controller, which preserves relation between modules and enables application of sequential processes.

As a result, a wide-range literature review covering both fundamental concerns and modern approaches has been performed and an extensive summary of this exploration is given in Chapter 2. Additionally, a new medical image processing and analysis framework has been developed and implementations of several filtering, segmentation and registration algorithms have been supplemented into the system. "*Implementation*" chapter gives detailed explanation on mapping physical and mathematical expressions into terms of computer programming. Technical information is supported with visual material in order to consolidate the understanding. Performance evaluation regarding the key parts of application has been performed and methods and quantitative results have been presented in "*Performance Evaluation*" chapter. Multiple medical image processing experiments have been done and results presented in relevant sections of "*Experimental Results*" chapter have been acquired using constructed application. Experiments have been designed in order to investigate input-output relations, and experimental results have been presented with both numerical and perceptual outcomes. Qualitative and quantitative discussions have been introduced for each result.

Resulting software system is a prototype application, which has capabilities of reading and writing DICOM images with metadata information, handling

sequential processes like image filtering, image restoration, image segmentation, image registration, and change detection over images acquired at different time instants. Additionally, maximum distances along dimensions, cross-sectional area (in 2D images), and amount of volume (in 3D images) can be computed for target region without any manual interaction that significantly affects the results.

Considering those capabilities, system can be respected as a substantial basis for an accurate, fast and robust automation system which can be utilized in decision making steps of diagnosis, prognosis, and therapy by the experts in field of medicine. Use of the system would definitely be helpful in reducing the amount of time and effort spent on common routines of radiology; consequently, reducing the rate of possible errors upcoming from excessive amount of manual interaction.

6.2. Future Work

Planned future expansions of this thesis study can be listed as follows:

- Filtering module will be enriched by appending implementations of various filter classes; such as, histogram-based filters, directional filters, and logarithm-based filters. Also, image regularization methods based on minimization of total variation will be implemented and plugged into the system.
- Segmentation module will be improved by designing a numerical solver which searches for the optimal input parameter sets for minimization of Mumford-Shah energy functional. Also, standard user presets will be prepared with intuitive selection of input combinations.

- Registration module will be developed to be used in registration of images belonging to body parts or structures other than brain. That requires implementation of deformable / non-global transformation models.
- Final version of the system is planned to be implemented in Java platform, adapting to common IEEE software standards, and optimizing performance issues.

REFERENCES

[1] Jiri Jan, *Medical Image Processing, Reconstruction and Restoration*.: CRC Press, 2005.

[2] Joseph V. Hajnal, Derek L. G. Hill, and David J. Hawkes, *Medical Image Registration*.: CRC Press, 2001.

[3] Samuel J Dwyer et al., "Medical Image Processing in Diagnostic Radiology," *Nuclear Science, IEEE Transactions on*, vol. 27, no. 3, pp. 1047 -1055, 1980.

[4] Barbara Zitova and Jan Flusser, "Image registration methods: a survey," *Image and Vision Computing*, vol. 21, no. 11, pp. 977 - 1000, 2003.

[5] Li Yi and Gao Zhijun, "A review of segmentation method for MR image," in *Image Analysis and Signal Processing (IASP), 2010 International Conference on*, pp. 351 -357, April 2010.

[6] G. C DeAngelis, I Ohzawa, and R. D Freeman, "Receptive-field dynamics in the central visual pathways.," *Trends in neurosciences*, vol. 18, no. 10, pp. 451--458, 1995.

[7] Richard A Young, "The Gaussian derivative model for spatial vision: I. Retinal mechanisms," *Spatial Vision*, vol. 2, pp. 273-293(21), 1987.

[8] Jean M Morel and Sergio Solimini, *Variational methods in image segmentation*. Cambridge, MA, USA: Birkhauser Boston Inc., 1995.

[9] Alexander H -D and Daisy T Cheng, "Heritage and early history of the boundary element method," *Engineering Analysis with Boundary Elements*, vol. 29, no. 3, pp. 268 - 302, 2005.

[10] Joachim Weickert, "A Review of Nonlinear Diffusion Filtering," in *Proceedings of the First International Conference on Scale-Space Theory in Computer Vision*, London, UK, pp. 3--28, 1997.

[11] P Perona and J Malik, "Scale-space and edge detection using anisotropic diffusion," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 12, no. 7, pp. 629 -639, 1990.

[12] Joachim Weickert, "Theoretical Foundations Of Anisotropic Diffusion In Image Processing," *Computing, Suppl*, vol. 11, 1996.

[13] G.-H Cottet and L Germain, "Image Processing through Reaction Combined with Nonlinear Diffusion," *Mathematics of Computation*, vol. 61, no. 204, pp. pp. 659-673, 1993.

[14] Stanley Osher and Leonid I Rudin, "Feature-Oriented Image Enhancement Using Shock Filters," *SIAM Journal on Numerical Analysis*, vol. 27, no. 4, pp. pp. 919-940, 1990.

[15] Luis Alvarez and Luis Mazorra, "Signal and Image Restoration Using Shock Filters and Anisotropic Diffusion," *SIAM Journal on Numerical Analysis*, vol. 31, no. 2, pp. pp. 590-605, 1994.

[16] G Gilboa, N Sochen, and Y.Y Zeevi, "Image enhancement and denoising by complex diffusion processes," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 26, no. 8, pp. 1020 -1036, aug. 2004.

[17] Guy Gilboa, Nir Sochen, and Yehoshua Zeevi,.: Springer Berlin / Heidelberg, vol. 2350, pp. 399-413, 2002.

[18] C Ludusan, O Lavialle, R Terebes, M Borda, and S Pop, "Image enhancement using hybrid shock filters," in *Automation Quality and Testing Robotics (AQTR), 2010 IEEE International Conference on*, vol. 3, pp. 1 -6, 2010.

[19] V.P Namboodiri and Subhasis Chaudhuri, "Shock filters based on implicit cluster separation," in *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, vol. 1, pp. 82 - 87 vol. 1, 2005.

[20] M.R Banham and A.K Katsaggelos, "Digital image restoration," *Signal Processing Magazine, IEEE*, vol. 14, no. 2, pp. 24 -41, 1997.

[21] Tony F Chan and Jianhong Shen, *Image Processing and Analysis: Variational, PDE, Wavelet, and Stochastic Methods*.: SIAM, 2005.

[22] C. R. Vogel and M. E. Oman, "Iterative Methods For Total Variation Denoising," *SIAM J. Sci. Comput*, vol. 17, 1996.

[23] Leonid I Rudin, Stanley Osher, and Emad Fatemi, "Nonlinear total variation based noise removal algorithms," *Phys. D*, vol. 60, pp. 259--268, 1992.

[24] T.F Chan, S Osher, and J Shen, "The digital TV filter and nonlinear denoising," *Image Processing, IEEE Transactions on*, vol. 10, no. 2, pp. 231 -241, 2001.

[25] P.L Combettes and J.-C Pesquet, "Image restoration subject to a total variation constraint," *Image Processing, IEEE Transactions on*, vol. 13, no. 9, pp. 1213 -1222, 2004.

[26] S Fu and C Zhang, "Adaptive non-convex total variation regularisation for image restoration," *Electronics Letters*, vol. 46, no. 13, pp. 907 -908, 2010.

[27] P Rodriguez and B Wohlberg, "An Iteratively Reweighted Norm Algorithm for Total Variation Regularization," in *Signals, Systems and Computers, 2006. ACSSC '06. Fortieth Asilomar Conference on*, pp. 892 -896, nov 2006.

[28] D. L Pham, C Xu, and J. L Prince,.: Annual Reviews, vol. 2, pp. 315--338, 2000.

[29] W E et al., "Utilizing segmented MRI data in image-guided surgery," *Int. J. Patt. Rec. Art. Intel*, vol. 11, pp. 1367-1397, 1997.

[30] David N Kennedy, Nikos Makris, Verne S Caviness Jr, and Andrew J Worth, "Neuroanatomical Segmentation in MRI: Technological Objectives," *IJPRAI*, vol. 11, no. 8, pp. 1161-1187, 1997.

[31] SM Lawrie and SS Abukmeil, "Brain abnormality in schizophrenia. A systematic and quantitative review of volumetric magnetic resonance imaging studies," *The British Journal of Psychiatry*, vol. 172, no. 2, pp. 110-120, 1998.

[32] P Taylor, "Computer aids for decision-making in diagnostic radiology--a literature review," *Br J Radiol*, vol. 68, no. 813, pp. 945-957, 1995.

[33] P. K Sahoo, S Soltani, A. K.C Wong, and Y. C Chen, "A survey of thresholding techniques," *Comput. Vision Graph. Image Process.*, vol. 41, pp. 233--260, 1988.

[34] S Beucher, "Watersheds of functions and picture segmentation," in *Acoustics, Speech, and Signal Processing, IEEE International Conference on ICASSP '82.*, vol. 7, pp. 1928 - 1931, 1982.

[35] L Vincent and P Soille, "Watersheds in digital spaces: an efficient algorithm based on immersion simulations," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 13, no. 6, pp. 583 -598, 1991.

[36] D Piraino, S Sundar, B Richmond, J Schils, and J Thome, "Segmentation Of Magnetic Resonance Images Using A Back Propagation Artificial Neural Network," in *Engineering in Medicine and Biology Society, 1991. Vol.13: 1991., Proceedings of the Annual International Conference of the IEEE*, pp. 1466 -1467, oct-3 # nov 1991.

[37] Haluk Derin, Howard Elliott, Roberto Cristi, and Donald Geman, "Bayes Smoothing Algorithms for Segmentation of Binary Images Modeled by Markov Random Fields," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. PAMI-6, no. 6, pp. 707 -720, 1984.

[38] Hui Huang and Jionghui Jiang, "Laplacian Operator Based Level Set Segmentation Algorithm for Medical Images," in *Image and Signal Processing, 2009. CISP '09. 2nd International Congress on*, pp. 1 -5, oct. 2009.

[39] T.F Chan and L.A Vese, "Active contours without edges," *Image Processing, IEEE Transactions on*, vol. 10, no. 2, pp. 266 -277, 2001.

[40] Michael Kass, Andrew Witkin, and Demetri Terzopoulos, "Snakes: Active contour models," *INTERNATIONAL JOURNAL OF COMPUTER VISION*, vol. 1, pp. 321--331, 1988.

[41] P.J Besl and R.C Jain, "Segmentation through variable-order surface fitting," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 10, no. 2, pp. 167 -192, 1988.

[42] V Duay, N Houhou, and J.-P Thiran, "Atlas-based segmentation of medical images locally constrained by level sets," in *Image Processing, 2005. ICIP 2005. IEEE International Conference on*, vol. 2, pp. II - 1286-9, sept. 2005.

[43] T Rohlfing and C.R Maurer, "Multi-classifier framework for atlas-based image segmentation," in *Computer Vision and Pattern Recognition, 2004. CVPR 2004. Proceedings of the 2004 IEEE Computer Society Conference on*, vol. 1, pp. I-255 - I-260 Vol.1, june-2 # july 2004.

[44] Min Chen, Shengyong Chen, and Qiu Guan, "Hybrid Contour Model for Segmentation of Cell Nucleolus and Membranes," in *Biomedical Engineering and Informatics, 2009. BMEI '09. 2nd International Conference on*, pp. 1 -5, oct. 2009.

[45] Y Ebrahimdoost et al., "Medical Image Segmentation Using Active Contours and a Level Set Model: Application to Pulmonary Embolism (PE) Segmentation," in *Digital Society, 2010. ICDS '10. Fourth International Conference on*, pp. 269 -273, feb. 2010.

[46] K Haris, S.N Efstratiadis, N Maglaveras, and A.K Katsaggelos, "Hybrid image segmentation using watersheds and fast region merging ," *Image Processing, IEEE Transactions on*, vol. 7, no. 12, pp. 1684 -1699, 1998.

[47] T Logeswari and M Karnan, "Hybrid Self Organizing Map for Improved Implementation of Brain MRI Segmentation," in *Signal Acquisition and Processing, 2010. ICSAP '10. International Conference on*, pp. 248 -252, feb. 2010.

[48] D Mumford and J Shah, "Optimal approximations by piecewise smooth functions and associated variational problems," *Communications on Pure and Applied Mathematics*, vol. 42, no. 5, pp. 577--685, 1989.

[49] MIT Press, *Visual Reconstruction*.: MIT Press, 1987.

[50] Stuart Geman and Donald Geman, "Stochastic Relaxation, Gibbs Distributions, and the Bayesian Restoration of Images," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. PAMI-6, no. 6, pp. 721 -741, 1984.

[51] Daniel Cremers, Florian Tischhauser, Joachim Weickert, and Christoph Schnorr, "Diffusion snakes: introducing statistical shape knowledge into the Mumford-Shah functional," *J. OF COMPUTER VISION*, vol. 50, 2002.

[52] Luminita A Vese and Tony F Chan, "A Multiphase Level Set Framework for Image Segmentation Using the Mumford and Shah Model," *International Journal of Computer Vision*, vol. 50, pp. 271-293, 2002.

[53] A Tsai, A Yezzi, and A.S Willsky, "Curve evolution implementation of the Mumford-Shah functional for image segmentation, denoising, interpolation, and magnification," *Image Processing, IEEE Transactions on*, vol. 10, no. 8, pp. 1169 -1186, 2001.

[54] Leo Grady and Christopher Alvino, *Reformulating and Optimizing the Mumford-Shah Functional on a Graph - A Faster, Lower Energy Solution*.: Springer Berlin / Heidelberg, vol. 5302, pp. 248-261, 2008.

[55] Xavier Bresson, Selim Esedoğlu, Pierre Vandergheynst, Jean-Philippe Thiran, and Stanley Osher, "Fast Global Minimization of the Active Contour/Snake Model," *Journal of Mathematical Imaging and Vision*, vol. 28, no. 2, pp. 151-167, 2007.

[56] Mila Nikolova, Selim Esedoglu, and Tony F Chan, "Algorithms for Finding Global Minimizers of Image Segmentation and Denoising Models," *SIAM Journal on Applied Mathematics*, vol. 66, no. 5, pp. 1632-1648, 2006.

[57] T Pock, A Chambolle, D Cremers, and H Bischof, "A convex relaxation approach for computing minimal partitions," in *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, pp. 810 - 817, june 2009.

[58] T Pock, D Cremers, H Bischof, and A Chambolle, "An algorithm for minimizing the Mumford-Shah functional," in *Computer Vision, 2009 IEEE 12th International Conference on*, pp. 1133 -1140, 292009-oct.2 2009.

[59] Antonin Chambolle, "Finite-differences discretizations of the mumford-shah functional," *M2AN*, vol. 33, no. 2, pp. 261-288, 1999.

[60] L.; Tortorelli Ambrosio, "Approximation of functional depending on jumps by elliptic functional via t-convergence," *Communications on Pure and Applied Mathematics*, vol. 999-1036, p. 43, 1990.

[61] J B and Max A Viergever, "A survey of medical image registration," *Medical Image Analysis*, vol. 2, no. 1, pp. 1 - 36, 1998.

[62] A Gholipour, N Kehtarnavaz, R Briggs, M Devous, and K Gopinath, "Brain Functional Localization: A Survey of Image Registration Techniques," *Medical Imaging, IEEE Transactions on*, vol. 26, no. 4, pp. 427 -451, 2007.

[63] D N Levin, C A Pelizzari, G T Chen, C T Chen, and M D Cooper, "Retrospective geometric correlation of MR, CT, and PET images.," *Radiology*, vol. 169, no. 3, pp. 817-823, 1988.

[64] G Borgefors, "Hierarchical chamfer matching: a parametric edge matching algorithm ," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 10, no. 6, pp. 849 -865, 1988.

[65] P.J Besl and H.D McKay, "A method for registration of 3-D shapes," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 14, no. 2, pp. 239 -256, 1992.

[66] Isaac N Bankman, *Handbook of medical imaging*. Orlando, FL, USA: Academic Press, Inc., 2000.

[67] Mark Holden et al., "Voxel similarity measures for 3-D serial MR brain image registration," *IEEE Transactions on Medical Imaging*, vol. 19, 2000.

[68] P Viola and W.M Wells, "Alignment by maximization of mutual information," in *Computer Vision, 1995. Proceedings., Fifth International Conference on*, pp. 16 -23, jun 1995.

[69] MathWorks. (2010, Jan.) MATLAB Image Processing Toolbox Documentation. [Online]. (Last Visited Date: 20/01/2011). http://www.mathworks.com/help/toolbox/images/ref/cp2tform.html

[70] T.M Lehmann, C Gonner, and K Spitzer, "Survey: interpolation methods in medical image processing," *Medical Imaging, IEEE Transactions on*, vol. 18, no. 11, pp. 1049 -1075, 1999.

[71] J. A Nelder and R Mead, "A Simplex Method for Function Minimization," *The Computer Journal*, vol. 7, no. 4, pp. 308-313, 1965.

[72] Jeffrey C Lagarias, James A Reeds, Margaret H Wright, and Paul E Wright, "Convergence Properties of the Nelder-Mead Simplex Method in Low Dimensions," *SIAM Journal of Optimization*, vol. 9, 1998.

[73] David W Shattuck, Gautam Prasad, Mubeena Mirza, Katherine L Narr, and Arthur W Toga, "Online resource for validation of brain segmentation methods," *NeuroImage*, vol. 45, no. 2, pp. 431 - 439, 2009.

[74] Laboratory of Neuro Imaging at UCLA. (2009) Segmentation Validation Engine. [Online]. (Last Visited Date: 20/01/2011). http://sve.loni.ucla.edu/

[75] PHILIPS Medical Systems. PHILIPS DICOM Image Server. [Online]. (Last Visited Date: 20/01/2011). ftp://ftp.philips.com/pub/

[76] Erkut Erdem, Aysun Sancar-Yilmaz, and Sibel Tari, "Mumford-Shah regularizer with spatial coherence," in *Proceedings of the 1st international conference on Scale space and variational methods in computer vision*, Berlin, Heidelberg, pp. 545--555, 2007.

APPENDIX A

NUMERICAL SOLUTION OF AMBROSIO-TORTORELLI MINIMIZATION TO MUMFORD-SHAH ENERGY FUNCTIONAL

Recalling Subsection 2.3.2.1 of Background, Ambrosio and Tortorelli proposed a minimization method for the Mumford-Shah energy, by replacing the edge-set term in Equation **(2.11)**, with a phase field energy term given by Equation **(2.12)**.

When Equation **(2.12)** is plugged into Equation **(2.11)**, Mumford-Shah energy functional is modified as below:

$$E_{AT} = \iint [\beta(u-z)^2 + \alpha|\nabla u|^2(1-v)^2 + (1/2)\{\rho|\nabla v|^2 + (v^2/\rho)\}]d\boldsymbol{x}.$$

(A.1)

Function v of Equation **(A.1)** can be expressed implicitly as given below:

$$\lim_{\rho \to 0}(1/2) \iint \{\rho |\nabla v|^2 + (v^2/\rho)\} = length(B). \quad \textbf{(A.2)}$$

Euler-Lagrange differential expressions for functions u and v can be implicitly given as:

$$\partial u/\partial t = \nabla((1-v)^2 \nabla u) - (\beta/\alpha)(u - z), \quad \textbf{(A.3)}$$

$$\partial v/\partial t = \nabla(\nabla v) - ((2\alpha\rho|\nabla u|^2 + 1)/\rho^2)(v - (2\alpha|\nabla u|^2)/(2\alpha|\nabla u|^2 + 1)). \quad \textbf{(A.4)}$$

Left-hand-sides of Euler-Lagrange equations empirically define the rate of change of the intensity values in signals u and v at specified points with respect to time, which - in this context - means the iteration number. Therefore these two equations give the characteristics of the evolution in the functions with increasing number of iterations.

u and v can be expressed in numerical form by replacing $\partial u/\partial t$ and $\partial v/\partial t$ with $(u_{i,j}^{k+1} - u_{i,j}^{k})/\Delta t$ and $(v_{i,j}^{k+1} - v_{i,j}^{k})/\Delta t$, respectively. These expressions are given by following equation:

$$u_{i,j}^{t+1} =$$

$$(1 + (\beta\Delta t/\alpha))^{-1}\{(\beta\Delta t/\alpha)z_{i,j}^{t} + u_{i,j}^{t} + (\Delta t/h^2)[d_{i+0.5,j}^{t+1}(u_{i+1,j}^{t} - u_{i,j}^{t}) - d_{i-0.5,j}^{t+1}(u_{i,j}^{t} - u_{i-1,j}^{t}) + d_{i,j+0.5}^{t+1}(u_{i,j+1}^{t} - u_{i,j}^{t}) - d_{i,j-0.5}^{t+1}(u_{i,j}^{t} - u_{i,j-1}^{t})]\}.$$

(A.5)

In Equation **(A.5)**, d terms define an interpolation of function v, as presented in Equations **(A.6)** and **(A.7)**:

$$d_{i\pm0.5,j}^{t+1} = (d_{i\pm1,j}^{t+1} + d_{i,j}^{t+1})/2\ , \qquad \textbf{(A.6)}$$

$$d_{i,j\pm0.5}^{t+1} = (d_{i,j\pm1}^{t+1} + d_{i,j}^{t+1})/2\ , \qquad \textbf{(A.7)}$$

where d is defined as:

$$d_{i,j}^{t} = \left(1 - v_{i,j}^{t}\right)^2. \qquad \textbf{(A.8)}$$

Numerical expression for function v can be presented as:

$$v_{i,j}^{t+1} = [1 + (\Delta t(1 + 2\alpha\rho A)/\rho^2)]^{-1}[(\Delta t/h^2)(v_{i+1,j}^{t} + v_{i-1,j}^{t} + v_{i,j+1}^{t} + v_{i,j-1}^{t} - 4v_{i,j}^{t}) + (\Delta t 2\alpha A/\rho^2) + v_{i,j}^{t}],$$

(A.9)

where,

$$A = \left|\nabla u_{i,j}^{t+1}\right|^2 = \left[\left(\left(u_{i,j+1}^{t+1} - u_{i,j-1}^{t+1}\right)^2 + \left(u_{i+1,j}^{t+1} - u_{i-1,j}^{t+1}\right)^2\right)/(2h)^2\right].$$

(A.10)

Programmatic implementation of the segmentation procedure is based on giving proper initial values to the functions u and v, and updating values by iterative sequential solution of these two signals. Details regarding implementation are introduced in Section 3.2. In preparation of this Appendix, derivations in reference [**76**] are made use of.